中国科学院大学
University of Chinese Academy of Sciences

硕士学位论文

引入双端历史信息和拼音信息改善神经机器翻译的研究

作者姓名：薛海洋

指导教师：冯洋 副研究员

中国科学院计算技术研究所

学位类别：工学硕士

学科专业：计算机软件与理论

培养单位：中国科学院计算技术研究所

2019 年 6 月

Research on Improving Neural Machine Translation with Bilingual History Information and Pinyin Information

A thesis submitted to the
University of Chinese Academy of Sciences
in partial fulfillment of the requirement
for the degree of
Master of Engineering
in Computer Software and Theory

By

Xue HaiYang

Supervisor : Associate Professor Feng Yang

Institute of Computing Technology, Chinese Academy of Sciences

June, 2019

中国科学院大学
学位论文原创性声明

本人郑重声明：所呈交的学位论文是本人在导师的指导下独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明或致谢。本人完全意识到本声明的法律结果由本人承担。

作者签名：

日　　期：

中国科学院大学
学位论文授权使用声明

本人完全了解并同意遵守中国科学院大学有关保存和使用学位论文的规定，即中国科学院大学有权保留送交学位论文的副本，允许该论文被查阅，可以按照学术研究公开原则和保护知识产权的原则公布该论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存、汇编本学位论文。

涉密及延迟公开的学位论文在解密或延迟期后适用本声明。

作者签名：　　导师签名：

日　　期：　　日　　期：

摘 要

机器翻译是自然语言处理中一项较为复杂的任务，传统的统计机器翻译包含许多自然语言处理中的子任务，包含分词，命名实体识别，词法分析，句法分析等。近年来，随着深度学习神经网络的兴起，神经机器翻译慢慢替代统计机器翻译，并取得了巨大的进步，在翻译的流畅度和忠实度上都有了极大提高。人们对于机器翻译的需求也越来越高，如何开发准确而流畅的翻译系统，成为当前研究的重点。

深度学习的进一步发展，使得自然语言处理上的众多任务的性能都得到了极大提升。在机器翻译领域，相比于传统的统计机器翻译方法，基于神经网络的机器翻译方法无需构建人工特征即可实现序列到序列的学习，深度学习的方法在模型表达能力和语言序列的表示上有着巨大的优势。本文主要研究基于神经网络的机器翻译，解决以下两个问题：一是通过对神经机器翻译中的注意力机制进行改进来解决当前翻译中的过翻译和漏翻译问题；二是对于语音翻译，引入拼音信息和篇章信息来对可能含有噪音的语音识别结果进行纠正，来解决语音翻译的噪音输入问题。通过对神经机器翻译的模型进行改进，旨在实现一个具有高鲁棒性高准确度的翻译系统。具体的研究内容如下：

1. 融合双端翻译历史信息的神经机器翻译

注意力机制 (Attention Mechanism) 极大地增强了神经机器翻译的性能，注意力机制的引入使得模型可以在解码阶段有选择性地获得源语言句子的信息来生成目标词。然而，可以发现在传统的基于注意力的神经机器翻译中，目标端和源端的已翻译历史信息都没有被充分利用，这常常导致在计算注意力向量时出现对齐错误，尤其是在一些复杂的情况下，神经机器翻译是无法预测正确的译文。为了解决这一问题，本文提出了一种新的注意力机制 Bilingual History Involved Attention，融合双端翻译历史的注意力模型。该注意力机制维护两个向量来跟踪目标端已生成的历史信息和源端已被翻译的历史信息。本文提出的方法在汉英翻译任务，在 NIST 测试集上取得了 1.4 BLEU 值的性能提升，并且与基线系统结果相比显著提高了对齐质量。

2. 引入拼音信息的机器翻译对于语音识别输入的鲁棒性改进

在许多实际应用中，神经机器翻译系统必须处理来自自动语音识别系统 (ASR) 的输入。语音识别的结果可能包含一些噪音，尤其是在复杂的环境中产

生的数据，经常会错误地生成一些词所对应的同音异形字和相近音异形字。这种情况往往会导致翻译性能急剧下降。在构建语音翻译系统时会产生两个很明显的问题，一是模型训练和系统测试时的不一致问题，二是由含有噪音的输入导致的翻译错误问题。本文创新性地提出了一种处理这两个问题的方法，以提高翻译时对语音识别错误输入的鲁棒性。首先对训练数据进行修改来模拟语音识别输出的错误类型，使训练和测试中的数据分布保持一致。其次，关注同音异形字和相近音异形字的自动语音识别错误，并利用它们的拼音信息帮助翻译模型从含有噪音的错误输入中纠正过来。在两个汉英数据集上的实验表明，该方法对语音识别噪音输入更具鲁棒性，并能显著优于强基线系统。

3. 基于拼音信息的篇章级机器翻译对于噪音输入纠错的研究

篇章信息一般是用于解决机器翻译中存在的一致性问题(包括指代，时态等)和歧义问题。篇章翻译能够利用丰富的上下文信息来改进当前句子的翻译效果。对于语音识别场景下的机器翻译，语音识别的结果往往是含有噪音的，从某种角度来看，语音识别的噪音输入也是一种歧义问题。基于此，本文创新性的利用篇章信息来对噪音输入进行纠错来提高模型泛化能力，结合篇章翻译的做法，从篇章信息中提取出与当前噪音词相关的信息，然后利用提取的信息来对当前词进行改进和信息补充。针对篇章翻译的特点和真实环境下的噪音问题，本文提出了基于篇章信息来提高神经机器翻译鲁棒性的方法，对整个系统的性能和泛化能力都带来极大的提高。

关键词： 深度学习，神经网络，机器翻译，注意力机制，鲁棒性

Abstract

Machine translation is a complex task in natural language processing. Traditional statistical machine translation involves a large number of subtasks, including word segmentation, named entity recognition, lexical analysis, syntactic analysis, etc. In recent years, neural machine translation has made great progress and gradually replaced statistical machine translation with the development of deep learning and neural networks. The demand of machine translation becomes more and more great. How to develop an accurate and smooth translation system has become the focus of current research.

With the further development of deep learning, the performance on many tasks in natural language processing has been greatly improved. In the field of machine translation, compared with the traditional statistical machine translation method, the neural machine translation method can realize sequence-to-sequence learning without constructing artificial features, and the deep learning method has great advantages in model expression and language sequence representation. This paper mainly studies the neural machine translation to solve the following two problems. One is to solve the problems of over-translation and under-translation by modifying the attention mechanism in neural machine translation. The second is to deal with the input from speech translation that may contain ASR noises. Pinyin information and document-level information are introduced to correct the wrong input. This paper aims to realize a translation system with high robustness and high accuracy. The specific research contents are as follows:

1. Neural Machine Translation with Bilingual History Involved Attention

The attention mechanism has greatly enhanced the performance of neural machine translation, since it can repeatedly and selectively read the representation of source sentence to generate target word. However, we found that historical information of both the target-end and the source-end were not fully exploited in conventional attention-base NMT, which often leads to misalignment and make wrong prediction, especially in sophisticated cases. To solve this problem, we propose a novel Bilingual History Involved Attention mechanism for text generation in NMT to tackle this problem, which maintains two vectors to keep track of both the target-end history and the source-end history. Our proposed approach achieves an improvement of 1.4 BLEU score in NIST Chinese-to-

English translation tasks and significantly improves alignment quality compared to baseline system.

2. Robust Neural Machine Translation with Pinyin Information for ASR Input

In many practical applications, the neural machine translation system must process the input from the Automatic Speech Recognition system (ASR). The result of speech recognition may contain some noises, especially the input generated in complex environment, in which correct words are often wrong substituted by homophones and words with similar pronunciation, often wrong to generate homophones and words with similar pronunciation, leading to a sharp decline in translation performance. There are two obvious problems in the construction of speech translation system: one is the inconsistency between model training and testing, the other is that translation errors caused by input noises. In order to improve the robustness of machine translation, this paper proposes an innovative method to deal with these two problems. First of all, we simulate the error types of speech recognition output in the training data, so that the data distribution in the training and testing is consistent. Secondly, we focus on the automatic speech recognition errors of homonyms and words with similar pronunciation, and use their pronunciation information to help the translation model to recover from ASR errors. Experiments on two Chinese-English data sets show that this method is more robust to speech recognition noisy input and can significantly outperform the strong baseline systems.

3. Research on Error Correction of Noise Input in Document-level Machine Translation Based on Pinyin Information

Document-level information is generally used to solve consistency problems (including anaphora, tense, etc.) and ambiguity problems in machine translation. Document-level translation can use rich contextual information to improve the translation of current source sentences. The results of speech recognition often contain noises in practical scenarios, and in some way, sentences with speech recognition errors is also an ambiguity problem. Inspired by document-level translation, we extract information related to noise words from document-level information, and then use the extracted information to improve annotations of the current words. Based on the characteristics of document-level translation and the problem of ASR noise input in practical environment, we propose a method to improve the robustness of neural machine translation based on document-level information, which greatly improves the performance and generalization ability of the

whole system.

Keywords: Deep Learning, Neural Network, Machine Translation, Attention Mechanism, Robustness

目 录

图形列表

表格列表

第 1 章　引言

1.1　研究背景及意义

随着全球化进程的不断加快，国与国之间的交流变得日益密切，跨语言的交流变得越来越迫切，机器翻译技术得到人们更多的关注。机器翻译在促进国家间文化交流，政治经济交融等方面的作用显得越来越重要。不同语言之间的信息交互已经成为了日常行为，而机器翻译带来了更加便利的沟通方式，便捷而有效。人们对于机器翻译的需求也越来越高，如何开发准确而流畅的翻译系统，成为当前各个研究机构的重点。

机器翻译（Machine Translation，MT）在分类上属于计算语言学。它的研究主要是使用计算机程序将句子或文章从一种自然语言翻译成另外一种自然语言。总的来说，机器翻译是一门集人工智能、计算机科学、语言科学与数理逻辑于一体的应用工程，其定位是跨学科或综合学科的一门技术。机器翻译技术的研究开始于 20 世纪 50 年代，由美国的研究人员率先提出，现在机器翻译的研究及使用几乎已经普及到世界上所有的国家。随着计算机技术的普及和计算机硬件性能的进一步提高，机器翻译程序的翻译能力也得到了提高。在经济水平较高的国家和地区，机器翻译技术不仅应用于文字的处理，还承担了翻译从业者的大部分日常翻译任务，而且正朝着智能化声控翻译通信技术的方向延伸。机器翻译以其低成本，翻译快速等优点成为翻译领域最流行的翻译工具。机器翻译是语言理解领域的经典测试，它包括两个部分: 语言分析和语言生成。大型的机器翻译系统有着巨大的商业用途，全球语言翻译是每年 400 亿美元的蓝海产业，而且每年还有非常不错的增速。尤其是近年来神经网络的发展，带给机器翻译非常广阔的发展前景，谷歌每天翻译超过 1000 亿个单词，Facebook 使用机器翻译来自动翻译不同语言的帖子和评论中的文本，打破了不同语言间直接的沟通障碍，让世界各地不同语言的人们能够顺畅交流。淘宝利用机器翻译技术实现了跨境贸易，并连接全球不同语言下的买家和卖家。

机器翻译是自然语言处理 (NLP，Natural Language Processing) 中的上游任务，是各种自然语言处理任务的集合，包括但不限于分词，词法分析 (lexical analysis)，句法分析 (syntactic analysis)，命名实体识别 (NER)，子词分割等，在技术上具有非常高的深度和广度，是当前自然语言处理领域最具有挑战性的任务。目前常用的机器翻译系统包括统计机器翻译系统 (SMT)(Brown 等, 1990) 和神经机器翻译

系统 (NMT)(Bahdanau 等, 2015)。

自 20 世纪 80 年代末以来，研究人员开始着力于对语音翻译技术 (Speech-to-speech Translation)(Kitano, 2012) 的研究，语音翻译又常被叫做口语翻译 (SLT, Spoken Language Translation)。简而言之，语音翻译是使计算机程序将从一种语言的语音转换到另一种语言的语音，相比于文本翻译，语音翻译实现的是语音到语音的翻译过程。它的基本思想是让计算机能够和人类一样在不同语言之间充当中间角色来转译。因为说话人所使用的语言在日常生活中普遍使用，人们也希望计算机翻译系统能够接受并实现任何口语句子的翻译，随着口语分析技术和语音识别技术的性能提高，这一希望也将变成现实。

语音翻译也是一门跨学科的研究方向，包括了语种识别，语音识别，口语合成，计算机技术，自然语言处理等多种学科，因此展开对语音翻译的研究具有非常重大的科学意义。另外，语音翻译还具有非常广阔的商业前景，该技术的突破可以被用于人类生活中的各个方面，例如国际贸易，国际民航信息咨询，旅游信息及时公布，国际会议和国际赛事信息综合服务。语音翻译拥有着相当大的潜力和巨大的社会效益。因此越来越多的国家投入巨大来开展语音翻译方面的工作，其中，德国联合教育研讨部 (BMBF) 在 1993 年到 2000 年期间领导并研究了 Verbmobil 语音翻译系统，主要是针对英、德、日等多语言进行自动语音翻译研究，先后投入高达 1.16 亿马克，大量的研究机构参与了 Verbmobil 的研究和开发工作，包括二十多所大学 (德国 Karlsruhe 大学等)，研究所 (美国 Stanford 大学的信息研究中心等)，和 7 个公司 (包括 GmbH，Philips 等)。之后的美国卡内基梅隆大学 (Carnegie Mellon University, CMU)，法国机器翻译研究所 (GETA-Clips)，意大利的科学技术研究所 (ITC-irst)、谷歌、苹果等世界著名大学、研究机构和科技企业都是语音翻译的重要推动者。

1.2 国内外本学科领域的发展现状

机器翻译是一个跨越性学科，它是建立在统计学，数学，计算机技术，语言学等多种学科基础上的。计算机科学技术的进步，语言学的进一步发展和概率统计的加入对机器翻译的理论基础和方法论研究产生了深远的影响，韦弗的机器翻译思想更是掀起了机器翻译的研究热潮。

乔姆斯基在 20 世纪 50 年代后期提出了短语结构语法，这一说法给出了“根据规则造句”的原则，但是短语结构语法使用单标记短语结构来描述句子的组成，描述粒度过于粗糙，因此存在约束能力较差和生成能力太强的问题。人们逐

渐了解到依靠单一的单词与句子结构信息并不能完全区分短语边界并确认短语类型。在很长的时间内，规则机器翻译都是当时主流的翻译技术。但基于规则却有着各种限制，构造语言规则需要巨大的人力成本，而且人工构造的规则往往之间会存在不可避免的矛盾。此外，规则方法在确保规则的完整性和适应性方面也存在很多不足。同时，统计方法在自然语言处理中的语音识别任务中取得了良好的性能，基于统计的机器翻译 (Brown 等, 1990) 随之获得了研究人员的关注。随着双语语料库的扩大和计算机硬件条件的提高，一种基于短语的机器翻译 (Koehn 等, 2003) 技术被提出。近年来，随着深度学习和服务器硬件的的迅猛发展，基于神经网络的机器翻译被研究者提出，神经机器翻译模型在各个翻译任务上的性能得到了很大提高，已经替代统计机器翻译成为主流翻译系统。

本章节，将梳理机器翻译的相关工作，描述机器翻译及其相关子任务的发展现状及其趋势，分析当前机器翻译系统存在的问题。

1.2.1 典型的机器翻译方法和技术

基于统计的机器翻译方法

除了一些特定的限制区域外，基于规则的机器翻译能够取得良好的效果，但在大多数实验中，基于规则的机器翻译还远不能满足人们的需求。随着平行语料库的扩展以及统计和信息理论在自然语言处理方向的应用，人们正试图将统计方法应用于机器翻译。对于机器翻译，我们可以从两个层面理解基于统计的方法，一种是在特定的机器翻译过程中应用某些概率方法，例如使用概率方法来解决词性标注和词义消歧的问题。另一种较为狭隘的理解是指纯粹基于统计结果的机器翻译，翻译所需的全部知识都来自平行语料库自身。IBM 的 Brown(Brown 等, 1990) 于 1990 年首次将刚开始应用于语音识别领域的统计模型，在法语到英语的机器翻译的模型上采用。基于词的统计机器翻译极大地推进的机器翻译的发展，但性能却因为建模单元受到很大限制，更多的研究人员开始着手基于短语的统计机器翻译 (Koehn 等, 2003) 方法。Moses(Koehn 等, 2006) 是当前维护比较好的开源统计机器翻译系统，由爱丁堡大学的工作人员维护开发。

神经机器翻译

2013 年，菲尔 · 布伦森和纳尔 · 凯尔纳提出了一种全新的端到端编码器-解码器结构 (Kalchbrenner 和 Blunsom, 2013)，并应用于机器翻译中。该模型可以通过使用卷积神经网络 (CNN) 作为编码器，将给定的源端文本输入编码成连续向量，然后使用循环神经网络 (RNN) 作为解码器，将连续状态向量转换为目标文

本，他们的提出的模型标志着神经机器翻译的诞生。神经机器翻译是一种全新的模型，是基于深度学习神经网络来对源语言映射到目标语言的方法，和线性的统计翻译翻译模型不同，神经机器翻译模型是非线性的映射，从而能够拟合更复杂的情况。神经机器翻译基于源语言和目标语言的的语义等价性，使用解码器和编码器来对状态向量进行非线性映射。随着深度学习相关技术的不断被提出，神经机器翻译 (NMT)(Bahdanau 等, 2015; Luong 等, 2015) 相比与统计机器翻译具有越来越多的优势，已经成为目前主流的机器翻译模型。

神经机器翻译源自于序列学习模型，其基本思想是通过神经网络来训练参数，能够直接实现源语言到目标语言的自动翻译。最早被研究者提出的编码器–解码器 (encoder-decoder) 模型是当前神经机器翻译模型的基础，编码器将源语言的输入文本编码为固定隐层向量，解码器读取隐层向量并按顺序依次生成目标句序列。为了解决源端语言句子过长而导致的长距离依赖问题，一般的方法是引入注意力机制 (Attention Mechanism) 来动态计算目标端词与源端词的权重。基于注意力机制神经机器翻译将源语言输入序列编码为隐层向量序列，当生成目标语言句子时，通过注意力机制动态关联与当前词相关的源语言隐层向量，从而大大提高神经机器翻译的表达能力，显着提高实验中的翻译效果。基于注意力机制的神经机器翻译不再将源端视为一个固定维度的向量，而是在解码过程中仅关注源语言句子的一部分，以此来增强捕获长句的能力。与普通的编码器–解码器模型相比，该方法在解码过程中集成了更多的源端信息，可以显着提高机器翻译的性能。

1.2.2 神经机器翻译流行框架

神经机器翻译是一个序列到序列的任务，当前翻译模型是基于编码器-解码器 (Encoder-Decoder) 框架，能够解决由一个可变长度的源端序列到另外一个可变长度的目标端序列的转换问题。即在编码阶段将整个源端输入序列进行编码得到源端隐层表示，在解码阶段模型最大化预测序列的概率，来对整个目标序列进行解码。目前流行的神经机器翻译流行框架包括 RNNSearch 模型和 Transformer 模型。

RNNSearch 模型

RNNSearch 模型最初由Bahdanau 等 (2015) 提出，它的编码器和解码器通过循环神经网络 (RNN) 来对源端序列和目标端序列进行建模，循环神经网络对拥有上下文联系的序列建模拥有着先天的优势。在翻译中，一个单词的语义可能和

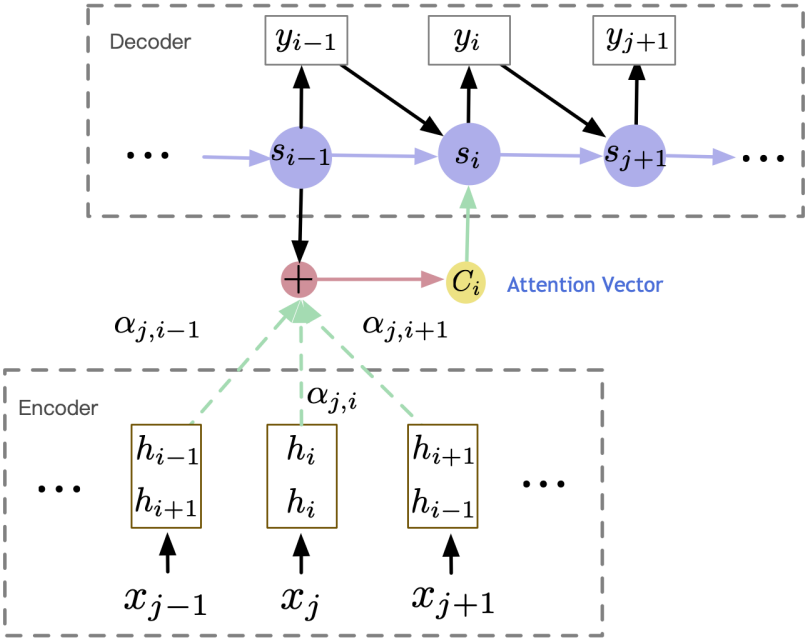

图 1.1 基于 RNNSearch 模型的机器翻译架构图

Figure 1.1 The RNNSearch model architecture

前后的单词相关联，也可能与之前的所有单词都相关联，借助循环神经网络中的记忆单元可以建立每个单词与其他词的联系。

RNNSearch 模型的编码器采用双向循环神经网络，双向循环神经网络在时间维度上按顺序和逆序处理输入序列，包括前向循环 (forward) 和后向循环 (backward)，并将每个时间步长的循环神经网络的输出进行拼接得到最终输出。经过双向的循环神经网络我们可以在每个时间步的输出节点都得到包括未来和过去的上下文信息。RNNSearch 模型的解码器同样使用循环神经网络来建模，和编码器不同的是，在解码时无法获得未来的信息，所以解码器是单向的循环神经网络。

和之前的编码器-解码器 (Encoder-Decoder) 框架不同的是，RNNSearch 模型引入了注意力机制 (Attention Mechanism)。编码器-解码器通过将源端输入序列编码成一个固定维度的向量，然后再将该向量由解码器解码，对于长句子序列来说，固定维度的向量是无法完整表达该序列的语义信息。注意力机制能够对编码后的所有上下文隐层向量进行解码，解码时到每个源端的距离都变为 $O(1)$，解决了固定维度的长序列语义表达问题。整个 RNNSearch 模型的架构我们可以在图1.1中了解到。

Transformer 模型

Transformer 模型同样是一个编码器-解码器 (Encoder-Decoder) 框架，最初由谷歌的Vaswani 等 (2017b) 提出。Transformer 模型相比于 RNNSearch 模型有了很大变化，使用注意力结构完全代替了循环神经网络来对编码器和解码器进行建模，抛弃了传统的编码器-解码器模型必须结合卷积神经网络或者循环神经网络的固有模式，只使用注意力机制。Transformer 模型不仅在翻译效果上提升非常大，在计算速度上也远远超越 RNNSearch 模型。

Transformer 模型只采用注意力机制的原因是考虑到循环神经网络 (包括长短期记忆网络等结构) 的建模必须是顺序进行的，也就是说循环神经网络的结构只能遵循从左向右或从右到左的顺序来依次计算，这种机制带来了两个问题:

* 时间步为 i 时刻的计算需要依赖时间步为 $i-1$ 时刻的计算结果，导致模型不具备并行能力;

* 顺序计算还存在了信息丢失的问题，尽管改进后的循环神经网络 (LSTM 等) 门控机制的结构能够缓解信息丢失问题，但对于长序列的依赖问题，循环神经网络并不能解决。

Transformer 模型完全使用了注意力机制能够非常好的解决上述问题，注意力机制在建模时任意位置的两个向量之间的距离是一个常量，不存在长期依赖问题；注意力机制不存在序列关系，可以更好的并行训练。

Transformer 模型有多层，每层由且仅由多头注意力和前馈神经网络组成。模型训练可以通过堆叠的形式进行搭建，当前 Transformer 模型训练的通用做法是通过搭建编码器和解码器各 6 层，总共 12 层的的结构来实现，具体 Transformer 模型的架构可由图1.2所示。

1.2.3 神经机器翻译相关技术

RNN 循环神经网络

长短时记忆单元 (LSTM)(Hochreiter 和 Schmidhuber, 1997) 和门控递归单元 (GRU)(Cho 等, 2014) 是目前 NLP 深度学习中，尤其是在序列任务中应用最广泛的模型。LSTM 和 GRU 的设计都是为了解决消失的梯度问题，通过门控机制来防止经常在标准 RNN 模型训练中出现的长期依赖问题。

GRU 是 LSTM 的标准 RNN 的两个变体，它们之间具有许多相同的属性，加入了门控单元和细胞单元。GRU 相比于 LSTM，将遗忘门 (forget gate) 和输入门 (input gate) 组合到一个更新门 (update gate) 中，还融合了单元状态 (cell state) 和隐藏状态 (hidden state)，并做了一些其他的更改。GRU 所得到的模型比标准的

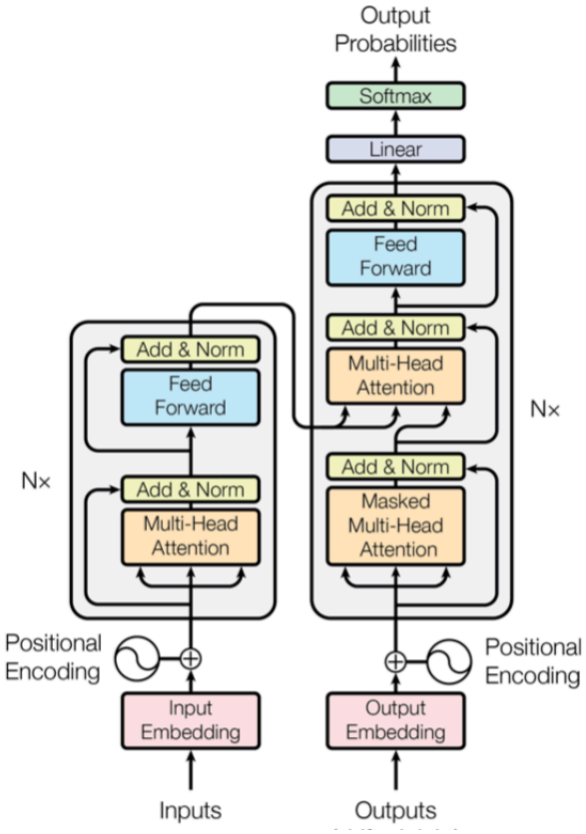

图 1.2 基于 Transformer 模型的机器翻译架构图

Figure 1.2 The Transformer model architecture

LSTM 模型简单，但在序列建模方面的性能与 LSTM 相当，且参数较少易于训练。

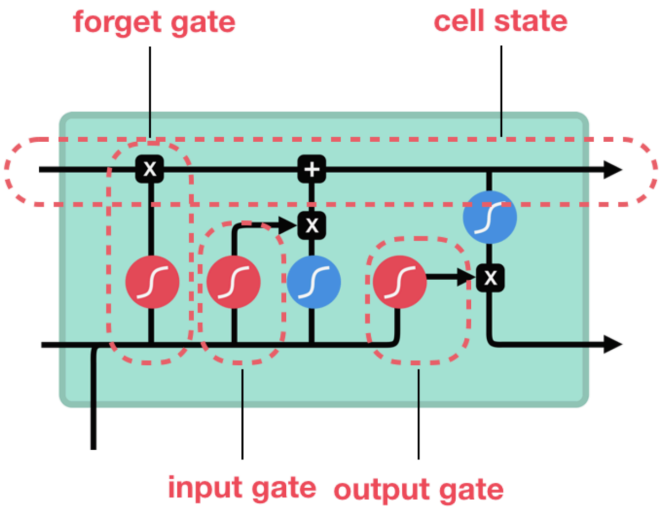

图 1.3 长短时记忆单元

Figure 1.3 Long Short Term Memory

LSTM

长短时记忆单元 (Long short-term memory, LSTM) 是一种改进的 RNN 模型，LSTM 早在 1997 年就被提出，主要目的是解决长序列训练时经常出现的梯度爆炸和梯度消失问题，并且相比标准 RNN，LSTM 还可以缓解长序列中的长期依赖问题。

LSTM 的优异表现得益于门控机制，我们在图1.3可以看到 LSTM 的结构，它包括三个门，分别是输入门，遗忘门和输出门和一个细胞单元：

$$\mathbf{i}_t = \sigma(\mathbf{x}_t\mathbf{U}^i + \mathbf{h}_{t-1}\mathbf{W}^i) \tag{1.1}$$

$$\mathbf{f}_t = \sigma(\mathbf{x}_t\mathbf{U}^f + \mathbf{h}_{t-1}\mathbf{W}^f) \tag{1.2}$$

$$\mathbf{o}_t = \sigma(\mathbf{x}_t\mathbf{U}^o + \mathbf{h}_{t-1}\mathbf{W}^o) \tag{1.3}$$

它们有完全相同的公式，只是参数矩阵不同，$\mathbf{W}$ 是前一隐藏层和当前隐藏层的线性连接，$\mathbf{U}$ 是将输入连接到当前隐藏层的权重矩阵。$\sigma(.)$ 是 sigmoid 函数函数，LSTM 中的门控机制其实是通过 sigmoid 函数将这些向量的值映射到 0 到 1 之间从而来充当门控信号，通过将它们与另一个向量进行点乘，我们可以决定想要“通过”多少该向量的信息。输入门定义了当前输入 $\mathbf{x}$ 要保留的状态，遗忘门决定了舍弃多少以前的状态 $\mathbf{h}_{t-1}$，若 $\mathbf{f}_t$ 为 1，则完全保留之前的状态，若为 0 则完全舍弃之前的状态。最后，输出门决定了希望向外部网络 (主要是更高层的网络和作为下一个时间步的输入) 输出多少内部状态。所有门的维度都是相同的。

$$\widetilde{\mathbf{M}}_t = \tanh(\mathbf{x}_t\mathbf{U}^g + \mathbf{h}_{t-1}\mathbf{W}^g) \tag{1.4}$$

$\widetilde{\mathbf{M}}_t$ 是根据当前输入和先前隐藏状态计算的“候选”隐藏状态。我们通过 $\widetilde{\mathbf{M}}_t$ 来计算当前步的细胞状态：

$$\mathbf{M}_t = \tanh(\widetilde{\mathbf{M}}_t) * \mathbf{o}_t \tag{1.5}$$

当前时刻的细胞状态是由通过遗忘门 $\mathbf{f}_t$ 控制的上一个时间步的细胞状态 $\mathbf{M}_{t-1}$ 和由输入门 $\mathbf{i}_t$ 控制的当前步的输入信息 $\widetilde{\mathbf{M}}_t$ 来决定的。

最终，当前时间步的隐层输出 $\mathbf{h}_t$ 由当前步的细胞状态 $\mathbf{M}_t$ 和输出门 $\mathbf{o}_t$ 计算得到：

$$\mathbf{h}_t = \sigma(\mathbf{f}_t * \mathbf{M}_{t-1} + \mathbf{i}_t * \widetilde{\mathbf{M}}_t) \tag{1.6}$$

GRU

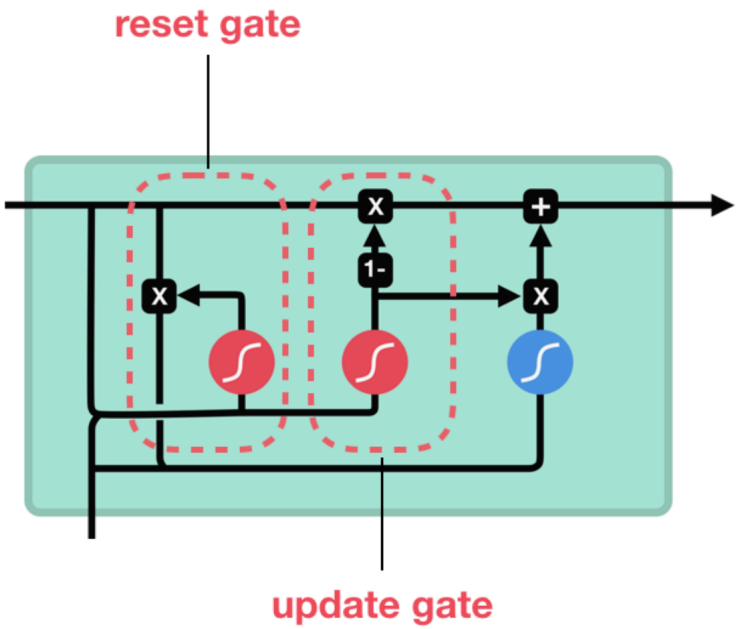

图 1.4 门控循环单元

Figure 1.4 Gated Recurrent Unit

门控循环单元 (Gate Recurrent Unit,GRU) 和 LSTM 一样，也是为了解决长句输入的长期依赖问题和反向传播中的梯度爆炸等问题而提出来的。相比 LSTM，使用 GRU 几乎能够达到相同的性能，而且 GRU 的参数更少，训练速度更快，我们在图1.4可以看到 GRU 的结构。

与 LSTM 最大的不同是，GRU 只有两个门，重置门 (reset gate) 和更新门 (update gate)

$$\mathbf{z}_t = \sigma(\mathbf{x}_t\mathbf{U}^z + \mathbf{h}_{t-1}\mathbf{W}^z) \tag{1.7}$$

$$\mathbf{r}_t = \sigma(\mathbf{x}_t\mathbf{U}^r + \mathbf{h}_{t-1}\mathbf{W}^r) \tag{1.8}$$

$$\tag{1.9}$$

更新门用于控制前一时间步的状态信息被带入到当前状态中的信息的多少，更新门 $\mathbf{z}_t$ 越大则表明前一时间步的状态信息保留的越多。重置门用于控制忽略前一时间步的状态信息的多少，重置门 $\mathbf{r}_t$ 越小说明忽略的状态信息越多。

和 LSTM 类似，然后计算候选隐藏层 $\widetilde{\mathbf{h}}_t$，候选隐藏层是当前时间步的新信息，其中 $\mathbf{r}_t$ 控制需要保留多少之前的信息，如果 $\mathbf{r}_t$ 为 0，那么 $\widetilde{\mathbf{h}}_t$ 只包含当前词的信息：

$$\widetilde{\mathbf{h}}_t = tanh(\mathbf{x}_t\mathbf{U}^h + ((\mathbf{r}_t * \mathbf{h}_{t-1})\mathbf{W}^h) \tag{1.10}$$

最后 $\mathbf{z}_t$ 控制需要从前一时间步的隐藏层 $\mathbf{h}_{t-1}$ 中遗忘多少信息，需要加入多少当前时间步的隐藏层信息 $\widetilde{\mathbf{h}}_t$，最终得到 $\mathbf{h}_t$:

$$\mathbf{h}_t = (1 - \mathbf{z}_t) * \mathbf{h}_{t-1} + \mathbf{z}_t * \widetilde{\mathbf{h}}_t) \tag{1.11}$$

注意力机制

自 2014 年Bahdanau 等 (2015) 的团队为神经机器翻译引入了注意力 (Attention) 机制之后，固定维度向量的问题也开始得到解决。注意力机制最早是在图像分类领域提出的 (Mnih 等, 2014)，能够让神经网络在训练时候更加能关注输入中更为相关的部分，而对于非相关部分投入更少的关注。受此启发，机器翻译在解码生成目标端词时候，根据语言间的对齐关系，源句子中只有很少的源端词与当前要生成的目标端词是相关联的，因此注意力机制是非常契合机器翻译任务的，我们可以使用注意力机制来对源语言句子动态的生成一个与当前解码词相关的加权向量，如图1.5所示，C 表示每一步的注意力向量。然后翻译系统能够根据这个加权的注意力向量而不是之前固定维度的向量来预测目标端词。注意力机制的应用使得神经机器翻译的翻译性能得到了非常显著提升，注意力-编码器-解码器的架构模型已经成为了机器翻译领最强的模型。

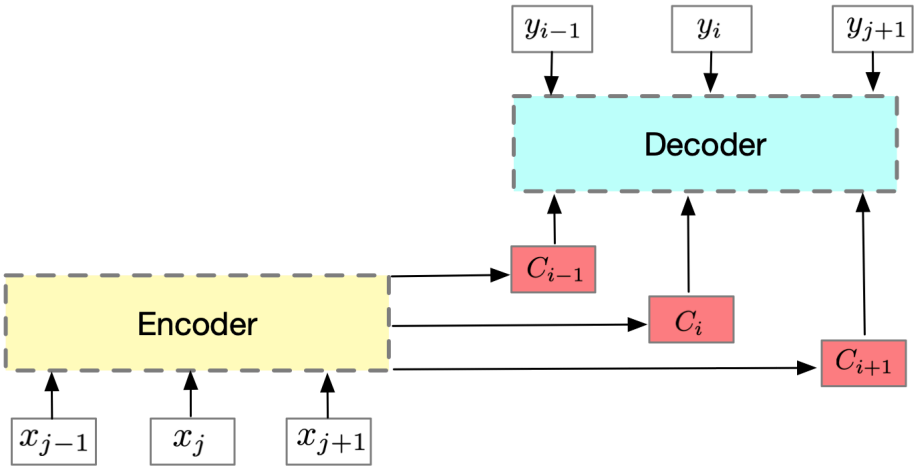

图 1.5 注意力机制结构图

Figure 1.5 The Attention Mechanism architecture

1.2.4 机器翻译的评价方法

BLEU 是一种评估文本质量的算法，经常被用于评价机器翻译和无偏差的人工翻译之间的对齐关系，BLEU 算法的核心思想是机器翻译的译文越接近人工翻

译，BLEU 得分越高，对应翻译译文质量就越好。因此 BLEU 算法计算的分数可以当做是评价机器翻译质量的一个重要指标。

评价一个翻译系统性能，最好的方式是通过专家对翻译译文进行人工评价，但是人工评价的方式速度非常慢，人工成本也非常高。同时人工评价极其依赖专业的翻译人员，但人工评价又往往存在主观性，而且对于长期大量的翻译评价任务，专业的翻译人员是无法满足需要的。为了解决这一问题，研究人员发明了一些机器翻译自动评价指标来代替人工评测，其中与人类评价最接近的就是 BLEU 算法。

引入 BP 值 (Brevity Penalty) 的 BLEU 值计算

当需要评价的翻译译文和任意一个参考译文的句子长度相等或超过时，BP 值设置为 1，当需要评价的翻译译文的长度比较短时，则用以下的算法计算 BP 值。用 c 来表示需要评价译文的句子长度，r 来表示参考译文的句子长度：

$$BP = \begin{cases} 1 \ \ if \ \ c > r \\ e^{1-r/c} \ \ if \ \ c \leq r \end{cases} \tag{1.12}$$

得到 BP 值之后，可以用以下算法计算 Bleu 值：

$$Bleu = BP \cdot \exp(\sum_{n=1}^{N} w_n \log p_n) \tag{1.13}$$

一般为了计算更加简便，使用对数来进行计算：

$$\log Blue = \min(1 - \frac{r}{c}, 0) + \sum_{n=1}^{N} w_n \log p_n \tag{1.14}$$

1.3 研究难点

基于注意力机制的神经机器翻译模型，由于对源端已翻译的历史信息和目标端已解码的信息利用不足，往往并不能很好的捕捉源端信息，就会导致过翻译 (如表1.1所示) 和漏翻译现象 (如表1.2所示)，过翻译是指某些词或短语会在翻译的文本中重复地出现，而漏翻译则是指有些词语没有得到有效的翻译。

对于语音翻译来说，尽管深度学习技术带来了全新的端到端的解决思路 (Serdyuk 等, 2018)，可以有效减少级联模式下的错误传播问题，但是这种方法对高质量的语音-翻译训练数据的需求远远大于机器翻译对于文本-文本训练数据的需求。目前可用的标注数据极其缺乏，已知的公开数据也不超过 300 个小时

源端输入	亚太地区将拥有比北美更多的 **电脑** 程式设计师和其他专业研发人员
翻译输出	... computer programmers and other professional **computer** researchers ...
参考译文	... computer programmers and other specialists ...

表 1.1 过翻译示例

Table 1.1 Example of over-translation

(Post 等, 2013; Kocabiyikoglu 等, 2018)，相比数千万的高质量双语文本语料，这些小规模的语音翻译数据远远不能满足实际需求，因此，端到端的语音翻译还没有成为主流的语音翻译方法。为了解决数据稀缺的问题，对于传统级联模式下的语音翻译，研究人员希望可以不用标准的双语平行语料，而是直接采用语音识别的结果训练机器翻译系统。然而，这和端到端语音翻译面临一样的数据缺乏问题。

由于神经机器翻译模型一般都是在高质量的双语平行数据上训练得到，而实际应用模型时，无法保证需要翻译的文本没有噪音或者错误，这就带来了模型训练和实际测试出现不匹配的问题，导致这个问题的主要原因在于模型训练时对训练数据过拟合，而对真实含有噪音的文本鲁棒性较差。

源端输入	**目前** 这些橄榄树暂时被储存
翻译输出	These olive trees are stored.
参考译文	These olive trees are **temporarily** stored.

表 1.2 漏翻译示例

Table 1.2 Example of under-translation

在实际场景中，用户的拼写错误或选词错误都会引入噪音，其中最严重的是在语音翻译场景。目前主流的语音翻译系统一般都包括语音识别和机器翻译两个独立的子系统，即采用先将语音识别成文本再翻译的级联方式。然而，由于环境噪音和用户口音等多种因素干扰，即使高性能的语音识别系统也经常出现识别错误，而在移动设备上离线使用的语音识别系统则识别错误更为严重，其识别性能较在线识别系统会有明显的下降。而作为下游应用的机器翻译严重依赖上游语音识别的准确度，含有噪音的语音识别结果往往导致神经机器翻译模型难以输出正确译文。因此，增强神经机器翻译对含有噪音输入的鲁棒性，对于增强

神经机器翻译的实际应用效果和改善用户体验具有重要的实用价值，同时也是一项极具挑战的研究课题。

正确输入	zhè fèn lǐ wù bǎo hán yī fèn shēn qíng 这 份 礼 物 饱 含 一 份 深 情
ASR 识别结果	zhè fèn lǐ wù bǎo hán yī fèn shēn qǐng 这 份 礼 物 饱 含 一 份 申 请
参考译文	This gift is full of affection
翻译译文	This gift contains an application

表 1.3 语音噪音输入的翻译示例

Table 1.3 An example of speech translation

本文鉴于神经机器翻译中面临的这些困难，拟解决以下问题：

* 对于历史信息利用不充分所导致的过翻译和漏翻译问题。

* 在机器翻译，尤其是语音翻译中，含有噪音的错误输入所导致的翻译错误问题。

1.4 本文的贡献

随着互联网的发展，不同语言不同国家的交流日益密切，机器翻译具有很广阔的应用前景。传统的机器翻译性能较差，无法满足人们的需求，神经机器翻译带来了翻译质量的巨大提升，但仍然有很大的提升空间，而且对于特定场景下的翻译需要更强的系统来保证翻译的鲁棒性。本文主要探讨了在神经机器翻译模型下，进一步提高模型能力并在特定的噪音环境下产生高质量的翻译译文。本文的主要工作包括：

1. 融合双端历史信息的神经机器翻译

神经机器翻译的注意力机制并不能很好的利用已翻译的源端信息和已经解码的目标端信息，因此本文提出全新的注意力架构 Bilingual History Involved Attention，将双端 (源端和目标端) 历史信息引入到注意力机制中。Bilingual History Involved Attention 模型不仅将新生成的目标词与源词对齐，而且还考虑先前生成的目标词和已翻译的源端词。模型直接模拟先前生成的目标词、已翻译的源端词与当前源词之间的依赖关系，其中每对源词和生成的目标词是一一对应的，旨在跟踪目标历史。该设计鼓励基于注意力的神经机器翻译系统考虑更多未翻译的源词。

2. 引入拼音信息的机器翻译对于语音识别输入的鲁棒性改进

针对汉英语音翻译任务，为了改进神经机器翻译系统对含有噪音的中文语音识别结果的鲁棒性，本文提出了一种鲁棒性更强的神经机器翻译模型训练方法，而不需要实际构造大量含有噪音的训练数据。基于 dropout 的思想，提出了一种正则化方法，常规方法主要应用于神经元，而本文提出的第一种方法则将这种思想应用于输入的单词本身，针对汉语语音识别结果，我们提出在模型训练过程中按照一定的概率 p 随机将源端句子中的汉字替换为噪音形式的同音异形字和相近音异形字，用来模拟实际的错误识别结果，从而增强神经机器翻译模型对于语言翻译等错误高发场景的翻译性能。另外，对于汉语当其中的噪音字和其对应的正确字是同音异形字和相近音异形字时，人们一般也可以正确理解这个含有噪音的句子，说明对于汉语来说，除了字形本身以外，其对应的拼音也能提供丰富的信息。通过对汉语语言特征的观察，我们还提出了引入汉语拼音作为附加特征的多特征输入表示方法，从而更好的提高神经机器翻译模型针对中文输入的鲁棒性。

3. 基于拼音信息的篇章级机器翻译对于噪音输入纠错的研究

这里的篇章严格来说并不是准确的篇章，其实是指当前输入的前几句，而非整篇文章。篇章信息一般是用于解决机器翻译中存在的一致性问题 (包括指代，时态等) 和歧义问题。篇章翻译能够利用丰富的上下文信息来改进当前句子的翻译效果。对于语音识别场景下的机器翻译，语音识别的结果往往是含有噪音的，从某种角度来看，语音识别的噪音输入也是一种歧义问题。基于此，本文创新性地利用篇章信息来对噪音输入进行纠错来提高模型泛化能力，结合篇章翻译的做法，从篇章信息中提取出与当前噪音词相关的信息，然后利用提取的信息来对当前词进行改进和信息补充。针对篇章翻译的特点和真实环境下的噪音问题，本文提出了基于篇章信息来提高神经机器翻译鲁棒性的方法，对整个系统的性能和泛化能力都带来极大的提高。

1.5 论文的组织

本文的组织结构如下：

第 1 章介绍了神经机器翻译的研究背景意义、相关的技术以及研究现状，并给出了本文的研究目标以及研究内容。

第 2 章提出了融合翻译系统源端和目标端的历史信息，包括源端已经被翻译的信息和目标端已经生成的历史信息，通过对传统神经网络的 Attention 机制进行改进，来提高整个翻译系统的对齐能力，解决现存模型过翻译和漏翻译的问

题。

第 3 章针对语音识别场景下的机器翻译，翻译系统的输入包含特定噪音，比如同音异形字，我们采用特定的采样方法来模拟 ASR 的输出，并将拼音作为一个有用特征来对可能错误的汉字进行改写，从而对系统起到纠错作用，极大提高系统的鲁棒性。

第 4 章同样是针对语音识别场景下的机器翻译，利用篇章信息来对含有噪音的输入进行修正，篇章翻译一般是用于解决一致性问题 (包括指代，时态等) 和歧义问题，具体来说，语音识别的噪音输入也是一种歧义问题。本文提出创新性的方法，借鉴篇章翻译的做法，对整个系统的性能和鲁棒性都带来极大的提高。

第 5 章对全文内容进行总结分析，并指出了进一步的研究方向。

第 2 章　融合双端历史信息的神经机器翻译

2.1　问题介绍

近年来，注意力模型 (Attention Mechanism) 在神经机器翻译 (Sutskever 等, 2014; **?**; Wang 等, 2016; Weng 等, 2017) 中占据了主导地位，并且取得了突破性进展，带来了神经机器翻译性能的飞跃提升。注意力机制主要包括传统的注意力模型 (Bahdanau 等, 2015)、覆盖模型 (the coverage model (Tu 等, 2016)、Transformer 模型 (Vaswani 等, 2017a) 等。与传统的统计机器翻译 (Chiang, 2005; Koehn 等, 2003; Zhai 等, 2012) 不同的是，由作者 Schuster 和 Paliwal提出的一种有代表性的基于注意力的神经机器翻译框架，将源句子采用 RNN 或双向 RNN 来编码成向量序列，然后是使用注意力机制动态对齐源端信息，并加权得到注意力向量，解码阶段使用另外一个 RNN 来生成目标句子。实验证明，注意力机制在神经机器翻译任务中取得了非常好的效果，这使得我们在训练时只需要较少的参数和训练数据就可以获得比较好的结果。动态对齐的注意力机制避免了使用固定维度的向量来表示整个源句子，大大提高了翻译准确率。

(1)	**源端输入**	人类共有 **23** 对染色体
	翻译译文	There were **23 23** pairs of chromosomes in human beings
(2)	**源端输入**	庆香港回归 **5 周年** 公务员书法大赛将举行
	翻译译文	Chinese civil service calligraphy competition to be held on Hong Kong’ s return

表 2.1 中到英翻译示例

Table 2.1 Examples of Chinese-English translation

然而，传统的注意力模型被设计用于预测目标词相对于源词的对齐，而没有考虑到目标词的生成可能与先前生成的目标词和翻译后的源词具有更强的相关性。RNN 在对较长的序列进行编码时，会受到长时记忆的影响，尽管门控循环单元 (GRU) 和长短时记忆网络 (LSTM) 相比于标准 RNN 来说，对于长序列的依赖问题有了较大的改进，但它们仍然不够完善，最近的研究也表明，源句越长神经机器翻译的表现越差。在这种情况下，注意力模型往往会倾向于忽略过去已近生成的目标端历史信息和翻译后的源端历史信息，从而导致对齐错误。

注意力机制的原理是首先根据当前解码目标端信息与源端句子中每个词的相关性来计算权重，然后根据权重来进行加权求和得到新的注意力向量作为每个解码时间步的源端信息。从这个过程中，可以看到在每一个时间步的注意力向量的计算只与当前的目标端信息和源端词的隐层表示有关，并不直接涉及到先前已经计算出的注意力向量，因此在不同的时间步长上是注意力向量的计算是相互独立的。而这么做导致的一个结果是注意力向量的计算是无法获得之前解码步中每个源端词的是否被翻译这一信号，从而导致过翻译和漏翻译 (Tu 等, 2016)。表 2.1提供了过度翻译和漏翻译漏翻译的示例，(1) 是一个过翻译的示例；(2) 是一个漏翻译的示例。示例 (1) 显示了“23”已被翻译两次的过翻译情况。如果模型在之前的解码步中能够得到源端词“23”已经被翻译的历史信息，那么在接下来的注意力计算时，“23”可能不会再被关注到。实例 (2) 说明了原文“5 周年”没有被翻译的漏翻译情况。如果模型可以得到源端词“5 周年”的一直没有被翻译到的信息，那么注意力的计算将进行调整，以给予“5 周年”更多的关注。因此，如果模型能够维持到当前解码步时与每个源端词是否被翻译的相关信息，源端每个词就可以得到更合理的注意。

基于注意力的对齐机制，为了解决过翻译和漏翻译的问题，本章提出了一种将双端历史信息引入注意力计算的方法 (bilingual history involve attention)。其主要思想是每次解码后，记录每个源端词是否已经被翻译的信息和已生成的目标端信息，然后利用 GRU 来累积与每个源词相关的源端和目标端历史信息。这样，就可以对每个源端词的被翻译的程度进行评估，并为注意力向量的计算提供更合理更有用的信息。在汉英和英德翻译任务上的实验表明，我们提出的方法在较强的基线基础上取得较高的改进，同时也可以产生更好的对齐结果。

2.2 相关工作

随着深度学习技术的普及与发展，机器翻译作为序列到序列的任务，也使用编码器-解码器的神经网络框架来对神经机器翻译模型进行建模。注意力机制的引入又给基于 RNN-编码器-解码器的神经翻译模型带来了性能上的巨大提高。近两年来，许多研究在基于注意力机制的神经机器翻译模型基础之上，提出了一些新的架构，从不同角度来对注意力机制进行改进，其中有一些工作 (Tu 等, 2016; Zhang 等, 2017) 将以前的注意力向量的历史信息融入到当前的注意力计算中，以便更好地对齐。

自注意力模型 (Self-Attention) 是最近研究中另一种比较流行注意力机制。

由Zhou 等 (2017) 提出的 Look-ahead Attention 能够对已生成的目标词之间的依赖关系进行建模。该模型通过参考之前生成的目标词信息来改进注意力机制，而以往的研究主要集中在与源词对齐的学习上。Lin 等 (2017) 进一步提出了一种可变的自注意力机制，提取句子的不同方面的特征信息，并将它们划分为多个向量表示。

为了提高神经机器翻译的记忆能力，Feng 等 (2017) 提出了一种新的记忆网络 (memory networks)，在神经机器翻译中引入了额外的存储单元，包括预设的外部知识和神经机器翻译模型本身学习了一些翻译模型。Cheng 等 (2016) 提出了一种具有外部共享内存的全新解码器，该解码器具有读、写和丢弃信息的能力。实际上，我们的模型可以看作是记忆网络的一个特例。

利用历史信息来提高注意力性能也是一种比较新颖的机制。Meng 等 (2016) 提出在注意力的计算中引入源端历史信息，在翻译过程中使用 Interactive Attention 可交互注意力机制来重写源端的隐层表示,Interactive Attention 通过读写操作来保持跟踪源端历史记录。Wang 等 (2018) 提出将目标端历史信息引入到注意力计算中，重点是对解码历史进行整合。然而，历史信息的利用基本上局限于源端或着目标端，我们的工作成功地将双端历史结合在一起，并取得了更好的效果。

2.3 系统背景

我们的模型是基于 RNNSearch 神经机器翻译模型 (Bahdanau 等, 2015) 来做的改进，主要是对注意力机制进行修改。基本框架是一个 seq2seq 的端到端系统，遵循解码器编码器架构，编码器由双向循环神经网络组成，产生源句的隐层向量表示。解码器与编码器类似，同样由循环神经网络组成，解码器同时读取对源端的隐层向量和已生成的目标端隐层向量来学习源端与目标端的对齐关系并预测翻译。框架中比较特殊的是该框架具有额外的注意力模块，这是一种改进源端和目标端对齐的机制，将在下一节中详细解释该模型及其各模块。

Encoder

基本的循环神经网络编码器是将源语言的句子压缩成一个固定维度的向量，那么在编码过程中序列中每个词的隐层状态其实只包含的前面词的信息。(Bahdanau 等, 2015) 使用双向的循环神经网络，来获得当前词的上下文，而非仅仅是

前文信息。系统的编码器采用双向 GRU 来得到源端句子中每个词的隐层表示：

$$\overrightarrow{\mathbf{h}}_j = \overrightarrow{\mathbf{GRU}}\left(x_j, \overrightarrow{\mathbf{h}}_{j-1}\right) \tag{2.1}$$

$$\overleftarrow{\mathbf{h}}_j = \overleftarrow{\mathbf{GRU}}\left(x_j, \overleftarrow{\mathbf{h}}_{j+1}\right) \tag{2.2}$$

这里 GRU 是一个双向的循环神经网络，从两个方向循环地对源端的输入进行编码，然后拼接每个词的输出状态，最终的源端隐层表示是将双向的 GRU 得到的隐层输出进行拼接：

$$\mathbf{h}_j = \left[\overrightarrow{\mathbf{h}}_j; \overleftarrow{\mathbf{h}}_j\right] \tag{2.3}$$

可以看到，$\mathbf{h}_j$ 结合了 GRU 前向和后向的表示，并且更加注意词 $\mathbf{x}_j$ 周围的单词，使得循环神经网络能够更好的表达当前输入的语义。得到的隐层序列 $(\mathbf{h}_1, \ldots, \mathbf{h}_j)$ 将用于后续的解码器和注意力机制中的对齐模型。

Attention

注意力机制的设计来源于人类的直觉，即在生成一个新词时，能够让神经网络在训练时候更加能关注输入中更为相关的部分，而对于非相关部分投入更少的关注。因此，模型更应该关注那些关联度高的源端词，忽视关联度低的词。注意力机制可以用来建立这些高度相关联的源端词和目标词之间的直接联系，然后翻译系统能够根据这个加权的 Attention 向量而不是之前固定长度的向量来预测目标端词。注意力机制的应用使得神经机器翻译的翻译性能得到了非常显著提升。

计算目标端词 $\mathbf{y}_j$ 与源端词 $\mathbf{h}_i$ 的联系时，使用前一步目标端词的隐层表示 $\mathbf{s}_{i-1}$ 与每个源端词的隐层表示 $(\mathbf{h}_1, \ldots, \mathbf{h}_j)$ 建立注意力关系，计算它们之间的对齐权重：

$$e_{ji} = \mathbf{v}_a^T \tanh\left(\mathbf{W}_a \mathbf{s}_{i-1} + \mathbf{U}_a \mathbf{h}_j\right) \tag{2.4}$$

对齐权重需要进行归一化，对公式2.4中得到的相关度进行 softmax 数值归一化，在第 j 个解码步的计算可得：

$$\alpha_{ji} = \frac{\exp\left(e_{ji}\right)}{\sum_{j'=1}^{l_s} \exp\left(e_{j'i}\right)} \tag{2.5}$$

最后，得到归一化后权重概率，权重概率越大表示与该源端词联系越紧密，将所有源隐藏状态与权重概率的加权和作为最终的注意力向量：

$$\mathbf{c}_i = \sum_{l=1}^{l_s} \alpha_{ji} \mathbf{h}_j \tag{2.6}$$

Decoder

定义目标端词 y_i 的条件概率为:

$$p(\mathbf{y}_i|\mathbf{y}_{<i}, \mathbf{x}) = f(\mathbf{y}_{i-1}, \mathbf{s}_i, \mathbf{c}_i) \tag{2.7}$$

其中 $\mathbf{s}_i$ 表示在时间步为 i 时的目标端一层状态，$\mathbf{c}_i$ 是时间步为 i 时的注意力向量，在介绍注意力机制时已经说明了注意力向量是如何求得的，接下来将对目标端隐层向量 $\mathbf{s}_i$ 的计算和最后的映射函数 $f(.)$ 进行说明。

解码器的工作原理是将生成的目标端隐层向量在目标词汇表中所有单词上映射一个概率分布，并以最大的概率输出目标单词。使用循环神经网络来对整个解码过程进行迭代，不断预测目标词。这里的循环神经网络是一个 GRU 的变体，可同时接收注意力向量作为输入。

所以当前步目标隐层状态 $\mathbf{s}_j$ 可以计算:

$$\mathbf{s}_i = f(\mathbf{y}_{i-1}, \mathbf{s}_{i-1}, \mathbf{c}_i) \tag{2.8}$$

最终映射的第 i 步的目标端词表概率 $\mathcal{D}_i$ 的计算主要是来源于前一步目标端词的词向量、当前步的注意力向量 $\mathbf{c}_i$ 和当前步的目标端隐层向量 $\mathbf{s}_i$ ，所以映射函数 $f(.)$ 为:

$$\mathbf{t}_i = g(\mathbf{y}_{i-1}, \mathbf{c}_i, \mathbf{s}_i) \tag{2.9}$$

$$\mathbf{o}_i = \mathbf{W}_o \mathbf{t}_i \tag{2.10}$$

$$\mathcal{D}_i = \text{softmax}(\mathbf{o}_i) \tag{2.11}$$

g 表示线性变换，$\mathbf{t}_i$ 可以通过一个参数向量 $\mathbf{W}_o$ 映射为 $\mathbf{o}_i$，最终单词的概率预测只与 $\mathbf{o}_i$ 的维度有关，$\mathbf{o}_i$ 的输出维度是目标端词表大小。

2.4 融入双端历史信息的注意力机制

注意力机制在每次解码时都计算当前解码词与每一个源端词的相似度值作为权重，并通过权重值有选择的收集源端语义信息得到注意力向量，然后解码器通过注意力向量得到预测的目标端译文。在这一过程中，注意力向量与预测的目

标信息之间存在语义映射的，这意味着加权和之后的源端信息是和生成的目标端信息在语义上是等价的。因此可以在每个解码步之后保留每个源端词被翻译程度的信息，并用注意力向量的归一化概率 α_{ji} 来表示，还可以得到已经生成的目标端信息。这样通过不断的解码就可以不断的累积每个源端词被翻译的信息和每一个已经生成的目标端信息，这种双端的历史信息可以很好地反映每个源词的翻译程度，应用到注意力机制的计算就可以得到更合理的注意力对齐概率。图3.3展示了我们方法的结构。在每生成一个目标端词 y_i 时，与每一个源端词 x_j

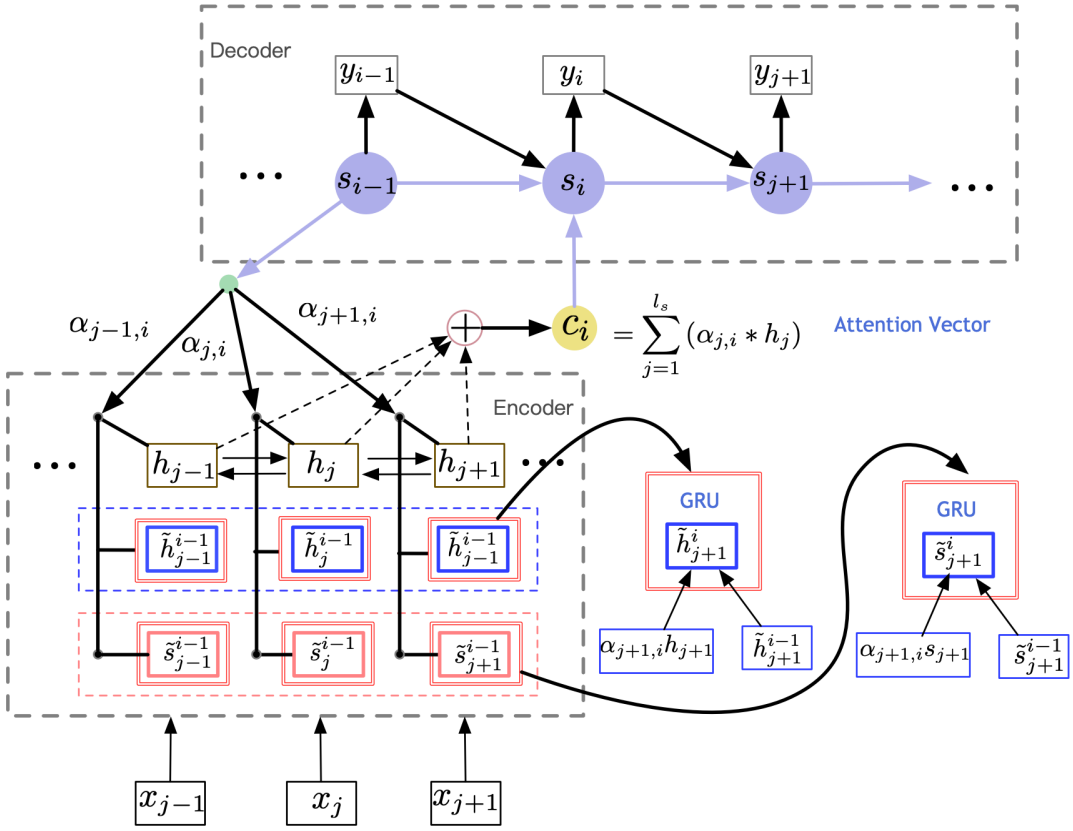

图 2.1 Bilingual History Involved Attention 架构图

Figure 2.1 Bilingual History Involved Attention architecture

相关联的源端信息就会通过一个 GRU 来累积，我们用 $\tilde{\mathbf{h}}_j^i$ 来表示这个累计的信息。值得说明的是，每个源端词都分别对应了一个累计向量。同样的与每一个源端词 x_j 相关联的目标端信息也会通过一个 GRU 来累积，使用 $\tilde{\mathbf{s}}_j^i$ 来表示。然后利用这些信息来生成下一个目标词。累积的双端信息具体的被用来辅助 Attention 的计算，并且在解码过程也会用到。

在这篇论文中，分别尝试使用不同端的历史信息来辅助注意力向量计算，来

验证方法的有效性：

* **引入源端历史信息的注意力机制 (SA-NMT)**: 只使用累积的源端信息来辅助计算注意力向量;

* **引入目标端历史信息的注意力机制 (TA-NMT)**: 只使用累积的目标端信息来辅助计算注意力向量;

* **融入双端历史信息的注意力机制 (BA-NMT)**: 同时使用双端的累积信息来辅助计算注意力向量。

2.4.1 引入源端历史信息的注意力机制

在第 i 个解码步，定义与每个源端词 x_j 所对应的源端历史信息是 $\tilde{\mathbf{h}}_j^{i-1}$。在生成目标端词 y_j 时，使用源端历史信息 $\tilde{\mathbf{h}}_j^{i-1}$ 来辅助注意力向量的计算：

$$e_{ji} = \mathbf{v}_a^T \tanh\left(\mathbf{W}_a \mathbf{s}_{i-1} + \mathbf{U}_a \mathbf{h}_j + \mathbf{V}_h \tilde{\mathbf{h}}_j^{i-1}\right) \tag{2.12}$$

然后可以通过公式 Eq. 2.5 和公式 Eq. 2.6得到注意力向量。

对于源端词 x_j 和注意力权重 α_{ji}，可以认为在第 i 个解码步时，x_j 所对应的已翻译的源端历史信息为：

$$\mathbf{I}_{ji}^S = \alpha_{ji} * \mathbf{h}_j \tag{2.13}$$

但是如果直接把每一步的 $\mathbf{I}_{ji}^S$ 直接相加，并不能真实地反映已翻译的源端历史信息，因为每一步解码后的 $\mathbf{I}_{ji}^S$ 与之前 $\mathbf{I}^S$ 是没有归一化的，模型使用 GRU 来动态更新累计已翻译的源端历史信息，通过 GRU 中的更新门和重置门能够得到更多有用的信息。更新如下：

$$\tilde{\mathbf{h}}_j^i = \mathbf{GRU}(\mathbf{I}_{ji}^S, \tilde{\mathbf{h}}_j^{i-1}) \tag{2.14}$$

另外，模型累计的已翻译的源端历史信息同样可以用来计算公式 Eq. 2.9中的 logit，这里通过加权求和的形式来集合各个 $\tilde{\mathbf{h}}_j^{i-1}$ 的信息，从而得到可用于求解 logit 的源端历史信息向量 $\tilde{\mathbf{h}}^{i-1}$:

$$\begin{aligned} \tilde{\mathbf{h}}^{i-1} &= \sum_j \alpha_{ji} * \tilde{\mathbf{h}}_j^{i-1} \\ \mathbf{t}_i &= g(\mathbf{y}_{i-1}, \mathbf{a}_i, \mathbf{s}_i, \tilde{\mathbf{h}}^{i-1}) \end{aligned} \tag{2.15}$$

2.4.2 引入目标端历史信息的注意力机制

在之前的章节我们曾提到在计算的注意力向量是源端信息的加权和，在语义上与目标端的隐层向量是等价的，因此同样可以对每一个源端词关联一个已

生成的目标端历史信息，并用这些信息辅助注意力向量的计算：

$$\mathbf{I}_{ji}^{T} = \alpha_{ji} * \mathbf{s}_{i-1} \tag{2.16}$$

同样的，每一步的 $\mathbf{I}_{ji}^{T}$ 没有归一化，模型仍然使用一个 GRU 来累积这些目标端历史信息：

$$\tilde{\mathbf{s}}_{j}^{i} = \mathbf{GRU}(\mathbf{I}_{ji}^{T}, \tilde{\mathbf{s}}_{j}^{i-1}) \tag{2.17}$$

$\tilde{\mathbf{s}}_{j}^{i}$ 表示已解码的目标端历史信息，同样使用 $\tilde{\mathbf{s}}_{j}^{i}$ 来辅助注意力向量的计算，因此对公式 Eq.2.4 的 attention model 进行改写：

$$e_{ji} = \mathbf{v}_{a}^{T} \tanh\left(\mathbf{W}_{a}\mathbf{s}_{i-1} + \mathbf{U}_{a}\mathbf{h}_{j} + \mathbf{V}_{s}\tilde{\mathbf{s}}_{j}^{i-1}\right) \tag{2.18}$$

$\tilde{\mathbf{s}}_{j}^{i}$ 可以表示第 i 步的目标端已翻译的历史信息与第 j 个源端隐层向量的关系，因此对公式 Eq.2.9中的 $\mathbf{t}_{i}$ 进行改写：

$$\begin{aligned} \tilde{\mathbf{s}}^{i-1} &= \sum_{j} \alpha_{ji} * \tilde{\mathbf{s}}_{j}^{i-1} \\ \mathbf{t}_{i} &= g(\mathbf{y}_{i-1}, \mathbf{a}_{i}, \mathbf{s}_{i}, \tilde{\mathbf{s}}^{i-1}) \end{aligned} \tag{2.19}$$

2.4.3 融入双端历史信息的注意力机制

图3.3说明了双端历史信息串联的结构，每个源端词所对应的双端历史信息表示了源端已经被翻译的历史信息的多少和目标端已经生成的历史信息的多少，而信息的多少反映了接下来的注意力向量计算是否要考虑该词。最后模型将双端历史信息来改写注意力机制的结构，辅助注意力向量的计算。可以得到以下公式：

$$\begin{aligned} e_{ji} = \mathbf{v}_{a}^{T} \tanh(&\mathbf{W}_{a}\mathbf{s}_{i-1} + \mathbf{U}_{a}\mathbf{h}_{j} + \\ &\mathbf{V}_{h}\tilde{\mathbf{h}}_{j}^{i-1} + \mathbf{V}_{s}\tilde{\mathbf{s}}_{j}^{i-1}) \end{aligned} \tag{2.20}$$

2.5 实验结果与分析

2.5.1 数据准备

本章实验主要是在 NIST 中英翻译任务上来验证模型性能，另外为了使实验更具说服力，本章实验在较大数据集英德翻译任务上也做了验证。所以实验数据集主要有两个：

NIST Zh→En: 中英数据集中包括 1.25M 双语语对[1]。选择 NIST 2002 作为验证集，包括 878 句双语对，选择 NIST 2003, 2004, 2005, 2006 作为测试集，分别包括 919, 1788, 1082，1664 句双语对。

WMT14 En→De: 在英德翻译任务上，使用 WMT14 作为训练集，包括 4.45M 英德双语语对，使用 newstest2013 作为的验证集，newstest2014 作为测试集。

在本章的实验中，使用 case-insensitive BLEU(Papineni 等, 2002) 来评价中英任务上的性能，case-sensitive BLEU 来评价英德任务。

2.5.2 对比系统

我们在以下系统做了实验，来对比我们模型的性能。

RNNSearch 采用传统基于注意力机制的神经机器翻译系统 (Bahdanau 等)，系统采用 PyTorch 框架[2]。

RNNSearch⋆ RNNSearch 系统的改进版，更多的细节可以通过这个链接了解[3].

NN-Coverage 针对 RNNSearch 的改进的模型 (Tu 等, 2016)，主要是维持了一个软覆盖向量，该向量包含一些历史信息，同样用于提高注意力机制的性能。

IA-Model 传统神经机器翻译模型的改进版，提出一个 Interactive Attention 来跟踪注意力向量的历史，并对源端信息进行改写。

2.5.3 系统配置

对于 NIST Zh→En 数据集，使用 16k 的 Byte Pair Encoding (BPE)(Sennrich 等, 2016) 操作数来进行数据集的字词分割，并用于源端和目标端。训练集中句子的最大长度是 128。对于 WMT En→De 数据集，BPE 的合并操作数被设置为 32k，训练集中句子的最大长度也是 128。

对所有的系统采用同样的系统配置，词向量维度大小设置为 512，编码器和解码器的隐层单元的维度也被设置为 512，所有参数的初始化都使用范围是 $[-0.1, 0.1]$ 的均匀分布 (uniform distribution)。优化算法采用 SGD(mini-batch stochastic gradient descent)，限制 batch 的大小为 4096 个字符。同时学习率的调整是通过 Adam 算法 (Kingma 和 Ba, 2015)($\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1e^{-6}$) 来优化的。

[1]双语对主要是从以下的数据集中抽取的 LDC2002E18, LDC2003E07, LDC2003E14, Hansards portion of LDC2004T07, LDC2004T08 and LDC2005T06

[2]http://pytorch.org

[3]https://github.com/nyu-dl/dl4mt-tutorial

Dropout rate 设置为 0.2，beam size 是 10。

Systems	**MT03**	**MT04**	**MT05**	**MT06**	**Average**
RNNSearch	35.75	38.68	34.69	37.61	*36.68*
RNNSearch*	42.03	44.58	42.33	42.40	*42.84*
NN-Coverage	42.69	44.92	42.74	42.79	*43.29*
IA-Model	42.83	45.14	42.94	43.12	*43.51*
Transformer-base	44.56	45.81	44.12	43.31	*44.45*
Our Method	**43.73**‡	**45.77**‡*	**43.58**‡*	**43.91**‡*	***44.25 +1.41***

表 2.2 中英翻译任务上的实验结果

Table 2.2 Results on Chinese-English translation

2.5.4 消融实验

本章中使用三种不同的方法来提高我们模型的能力，例如，我们动态的维持一个向量，能够保存源端的历史信息，并且用它来辅助注意力向量的计算，模型能够利用这种机制来解决漏翻译的问题。另外还对每一个源端词与已生成的目标端信息建立一对一的联系，这种目标端的历史信息是能够提供哪些目标端词是已经被翻译出来的，这在接下来的解码会避免再生成已翻译的词。

Systems	**Zh→En**
RNNSearch	36.68
RNNSearch*	42.84
+ SA-NMT	43.52
+ TA-NMT	43.83
+ BA-NMT	**44.25**

表 2.3 中英翻译任务各模块模型对比得分

Table 2.3 Results of each module of the Chinese-English translation

我们模型在中英翻译任务的测试集上的结果在表 Table 2.3列出，很明显可以看到本章提出的 History Involved Attention Model 模型在所有的情况下都超过了基线系统 RNNSearch* 系统。其中，只采用引入源端历史时，模型取得了 43.52

Systems	En→De
RNNSearch	25.76
+ SA-NMT	26.11
+ TA-NMT	26.32
+ BA-NMT	**26.58**

表 2.4 英德翻译任务各模块模型对比得分

Table 2.4 Results of each module of the English-German translation

BLEU 的结果，这比 RNNSearch* 提高了 0.68 BLEU，同样地只采用引入目标端历史时，我们模型相比基线系统有了 0.99 BLEU 的提高。最终将两个系统进行融合，取得了 1.41 BLEU 的明显提高，这说明我们的模型确实能够有效地改进现有的基线模型。

英德任务的结果可以在表 Table 2.4得到，改进系统 BA-NMT 同样展现出优势，相比 RNNSearch* 有 0.8 BLEU 的提高。鉴于中英和英德两个任务上的结果，可以得出结论，BA-NMT 确实能够更好地利用双端历史信息，提高翻译性能。

2.5.5 对齐质量分析

在第一章的介绍部分，我们用两个例子 Table.3.1来说明了传统神经机器翻译中过翻译和漏翻译的现象，与之对应的是我们的 BA-NMT 模型在同样的示例中表现地非常好，在这里我们展示了过翻译和漏翻译的对齐图.2.2，图 (1) 和 (2) 来自于 RNNSearch*，图 (3) 和 (4) 来自我们的模型 BA-NMT，可以直观的对比 BA-NMT 模型和基础的 Attention-NMT 在过翻译和漏翻译的示例中的对齐，颜色较深表示对齐概率更高，可以看到在 BA-NMT 模型的对齐图，一旦一个源端词被翻译了，在接下来的翻译，该词对应的图块的颜色一直是比较浅的，这说明这个词的被翻译的历史信息被考虑到，在接下来的对齐便不会再被考虑，比如说第一个源端词“庆”。

为了更好地说明 BA-NMT 模型在对齐上的优越表现，我们在 BLEU 评分的结果证明我们的方法可以实现更准确的翻译，但是大家普遍认为，更好的翻译应该具有更高质量的对齐，同时我们模型也是为了解决过翻译和漏翻译问题，本质上也是提高对齐质量，所以为了更好地说明 BA-NMT 模型的能力，我们尝试用 AER (Och, 2003) 这个指标来评价模型。对于数据集，采用了来自Liu 和 Sun (2015) 的对齐数据集，其中包含 900 句的中英对齐句子，来评估了对齐的质量。

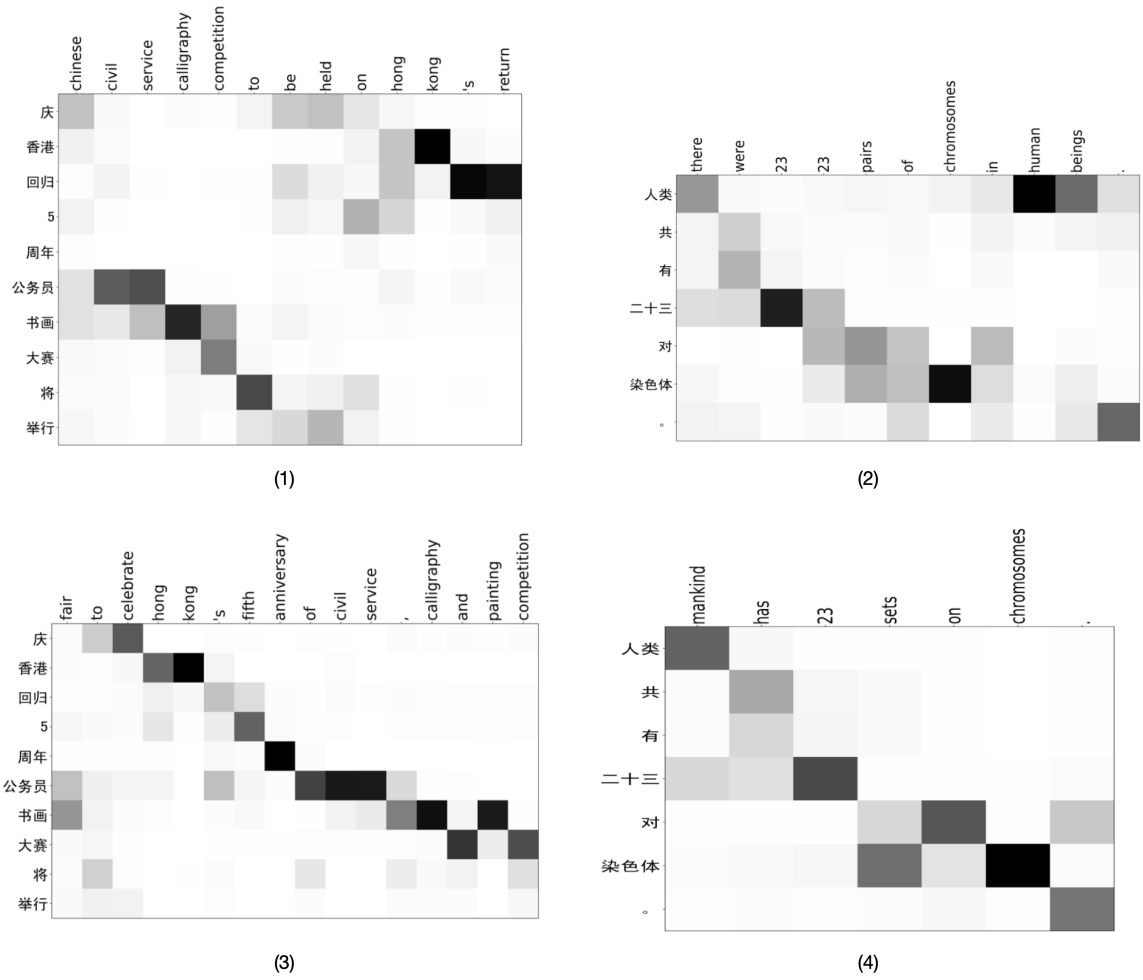

图 2.2 中英翻译示例对齐图

Figure 2.2 Alignment diagram of examples in Chinese-English translation

结果见表 Table2.5, 相比于 RNNSearch* 系统，我们的 BA-NMT 系统 AER 得分更低，对齐质量更好。

SYSTEMS	BLEU	AER
RNNSearch*	42.84	44.03
Our Method	**44.25**	**42.16**

表 2.5 中英翻译任务上的对齐质量对比

Table 2.5 Alignment quality in Chinese-English translation

2.6 本章小结

在这项工作中，展示了一个引入双端历史的注意力模型 (Bilingual History Involved Attention)，这是一个基于基线注意力机制的改进模型。我们模型的核心创新之处在于模型能够同时保持对目标历史信息和源端历史信息的动态跟踪与更新，这有利于模型拥有更好的双端词对齐，生成更准确的翻译。本章还探索了模型在神经机器翻译任务中的应用，并对将历史信息集成到神经机器翻译中的三种方法进行了实验。实验的结果也与预期是一致的，这证明我们的 Bilingual History Involved Attention 模型能够获得比基线模型更好的对齐质量，尤其是在一些复杂的情况。这种特性可以有效地缓解过翻译和漏翻译的现象。

第 3 章　引入拼音信息的机器翻译对于语音识别输入的鲁棒性改进

3.1　问题介绍

神经机器翻译已经取得了突破性进展，流畅性与忠实度相比于传统的统计机器翻译有了巨大提升。然而，由于神经机器翻译模型的训练都是在质量较高的双语语料上得到的。而我们在实际生活中，并不能保证翻译输入的文本是没有错误或者噪音的，这就带来了模型训练和模型测试不一致的问题 (Ruiz 等, 2015)，从而导致对于质量不高的输入的翻译结果会比较差，这个问题出现的主要原因还是在于模型训练时对高质量的训练数据产生了过拟合，而对含有噪音真实场景下的输入鲁棒性较差。

Gold input	这	份	礼	物	饱	含	一	份	**深**	**情**.
	zhè	fèn	lǐ	wù	bǎo	hán	yī	fèn	**shēn**	**qíng**.
ASR-HM	这	份	礼	物	饱	含	一	份	**申**	**请**.
	zhè	fèn	lǐ	wù	bǎo	hán	yī	fèn	**shēn**	**qǐng**.
ASR-SP	这	份	礼	物	饱	含	一	份	**心**	**情**.
	zhè	fèn	lǐ	wù	bǎo	hán	yī	fèn	**xīn**	**qíng**.
Reference	This gift is full of affection.									
Trans-HM	This gift contains an application.									
Trans-SP	This gift is full of mood.									

表 3.1 包含语音识别错误的中英翻译示例

Table 3.1 Examples of English-Chinese translation with speech recognition errors

对于语音翻译来说，目前主流的系统一般都包括两部分，一个是语音识别系统另一个是机器翻译系统，两个系统相互独立，先将语音识别成文本，再将文本输入机器翻译系统翻译 (Matusov 等, 2006)。然而，由于语音识别系统性能的限制和复杂的应用环境，语音识别结果经常是包含噪音的。而作为终端系统的机器翻译严重依赖一个高质量无噪音的输入，含有噪音的语音识别文本往往使翻译结果准确度大大下降 (Weiss 等, 2017; Cho 等, 2017)。因此，提高神经机器翻译对含有噪音输入的鲁棒性，具有非常大的实用价值。

在实际语音翻译环境中，用户的语音往往会被识别为同音异形字和相近音异形字，导致语音识别结果作为输入时是包含噪音的，使得机器翻译输出错误的译文，从表 Table 3.1(“ASR-HM” 是同音异形字错误，”ASR-SP” 是相近音异形字

错误, “Trans-HM” 和 “Trans-SP” 分别表示对应的机器翻译译文), 可以看出同音异形字和相近音异形字的识别错误对机器翻译带来了非常负面的影响。在本章节，我们提出了相近音异形字错误修正方法 (SP-Amendment) 和同音异形字错误修正方法 (HM-Amendment)，来解决上述问题，最终实验表明，我们提出的方法极大的增强了神经机器翻译在噪音输入下的鲁棒性。

3.2 相关工作

在语音翻译的场景下，语音识别系统会将识别错误的文本传输到下游神经机器翻译系统中，因此提高神经机器翻译的鲁棒性是非常有必要的。之前的一些工作试图通过将真实场景下语音识别输出作为训练翻译系统的语料来拟合噪音 (Peitz 等, 2012; Tsvetkov 等, 2014)。实验表明这种方法在一定程度上确实可以拟合含有噪音的错误输入 (Serdyuk, Wang, Fuegen, Kumar, Liu, and Bengio, 2018; Berard, Besacier, Kocabiyikoglu, and Pietquin, 2018)，但遗憾的是，这种真实场景下的噪音训练数据是非常少且难以获得的，相比之下，我们的方法利用了大规模的高质量平行语料库。

最近 Sperber 等也将语音识别中的噪声输出作为训练集训练机器翻译模型，模型在训练过程中引入了人为制造的噪音，从实验结果来看只取得了很小的改进，但却对于无噪音文本的翻译效果急剧下降，我们的方法不仅显著增强了神经机器翻译模型对噪声测试集的鲁棒性，而且还在干净的测试集上保证了性能的稳定。

在机器翻译领域，Cheng 等也提出了一种类似的方法，他们提出了两种构造对抗样本的方法，这两种对抗样本是在原本数据集上做了微小的扰动，并且通过编码器和解码器对输入的扰动句子和其原始句子同时训练来达到两中输入的相似保持一致，来以此使得机器翻译模型具有更强的鲁棒性。与之相比，我们的方法有以下几个优点：1) 我们构造噪声样本的方法是有效的，但在训练时不需要复杂的词语相似度计算，2) 我们的方法只有一个超参数，而不需要花费太多的精力进行性能调优，3) 我们方法的训练是有效的，不需要预训练神经机器翻译模型和复杂的鉴别器 (Discriminator)；4) 在不同程度的噪音环境，我们的方法对噪声输入具有稳定的性能。

我们的方法受到神经机器翻译与汉语言输入特征相结合的工作的启发 (Sennrich 和 Haddow, 2016)。汉语的语言特征，如部首和拼音，已被证明对来自中文的神经机器翻译 (Zhang 和 Matsumoto, 2017; Du 和 Way, 2017) 和中文语音识

别 (Chan 和 Lane, 2016) 是有帮助的。我们还将拼音作为一种额外的输入特征加入到神经机器翻译模型中，旨在进一步提高神经机器翻译的鲁棒性。

3.3 系统背景

我们的模型是基于 Transformer 模型 (Self-Attention based neural machine translation model) (Vaswani 等, 2017b) 做的改进。Transformer 模型同样也是一个 Encoder-Decoder 的结构，但它完全舍弃了传统翻译模型中的 CNN 和 RNN 结构，整个模型结构完全是由注意力机制构成，准确来说，Transformer 模型由且仅由自注意力 (Self-Attenion) 和前馈神经网络 (Feed Forward Neural Network) 组成。接下来我们将对 Transformer 模型进行详细介绍。

3.3.1 编码器

编码器由 6 个完全相同的层 (layer) 组成，每个层由两个子层 (sub-layer) 组成，分别是多头自注意力层 (Multi-Head Self-Attention) 和一个简单的前馈神经网络层 (Position-wise feed-forward networks)。在子层之间使用残差网络 (Residual connection)，接着进行层标准化 (Layer normalization)。

残差连接计算可以表示为：

$$\mathrm{H}(\mathbf{x}) = \mathrm{F}(\mathbf{x}) + \mathbf{x} \tag{3.1}$$

$\mathrm{F}(\mathbf{x})$ 是我们定义的函数，因此这里可以将子层输出表示为：

$$\mathbf{so}_x = \mathrm{LayerNorm}(\mathbf{x} + \mathrm{Sublayer}(\mathbf{x})) \tag{3.2}$$

$\mathbf{so}_x$ 表示子层输出。接下来我们将介绍一下这两个子层。

多头注意力机制

在传统的神经机器翻译中，注意力机制可以由以下形式来表示：

$$\mathbf{c} = \mathrm{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) \tag{3.3}$$

$\mathbf{Q}$，$\mathbf{K}$，$\mathbf{V}$ 分别表示查询矩阵，键矩阵和值矩阵。

与传统的注意力机制相比，多头注意力模型不再是单一的查询、键、值，而是拥有 h 个不同的查询、键、值，结构如图3.1所示，多头注意力模型通过 h 个不同的线性变换对 $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ 进行投影：

$$\mathbf{Q}^i, \mathbf{K}^i, \mathbf{V}^i = \mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V, i \in [1, h] \tag{3.4}$$

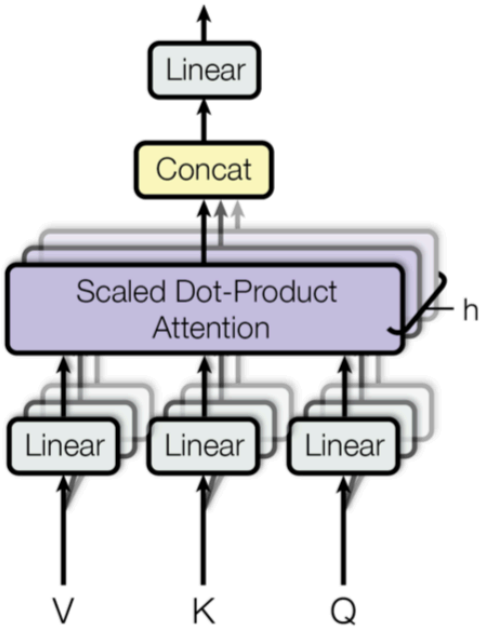

图 3.1 多头注意力模型架构图

Figure 3.1 Mulit-head Attention architecture

$\mathbf{Q}^i, \mathbf{K}^i, \mathbf{V}^i$ 分别是查询、键、值在第 i 个 head 的向量表示，$\mathbf{W}_i^Q, \mathbf{W}_i^K, \mathbf{W}_i^V$ 是映射矩阵，h 是 Attention head 的数量，对每个头分别做注意力计算：

$$\mathbf{head}_i = Attention(\mathbf{QW_i}^{\boldsymbol{Q}}, \mathbf{KW_i}^{\boldsymbol{K}}, \mathbf{VW_i}^{\boldsymbol{V}}) \tag{3.5}$$

最后将不同的注意力函数计算的结果拼接起来：

$$\text{MulitHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\mathbf{head}_1, \ldots, \mathbf{head}_h)\mathbf{W}^O \tag{3.6}$$

这是标准的多头注意力模型，而多头自注意力模型 (mulithead Self-Attention) 中的 $\mathbf{Q}$，$\mathbf{K}$，$\mathbf{V}$ 相同。在图3.2所示，在 *Attention* 函数的具体计算中，Transformer 模型采用归一化的点乘注意力 (scaled dot-product):

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}(\frac{\mathbf{QK}^T}{\sqrt{d_k}})\mathbf{V} \tag{3.7}$$

相比于之前的注意力机制采用的 Additive Attention 的方法，在 d_k 比较小的时候，达到的效果和点乘注意力类似，但当 d_k 比较大的时候，不进行缩放表现反而更好，但点乘的计算速度更快，在缩放之后会将影响降到最小。

位置全连接前馈网络

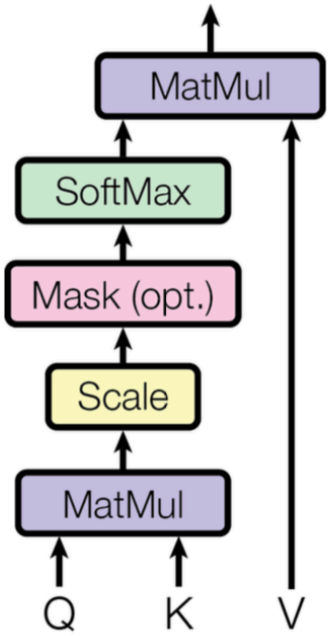

图 3.2 归一化的点乘注意力图示

Figure 3.2 Scaled dot-product attention

第二个子层是位置全连接前馈网络 (Position-wise feed-forward networks, FFN)，本质是多层感知器 MLP 的变形。位置全连接前馈网络层有两层，第一层的激活函数是 ReLU，第二层是一个线性激活函数，如果 multi-head 输出表示为 $\mathbf{z}$，则 FFN 可以表示为:

$$\mathrm{FFN}(\mathbf{z}) = max(0, \mathbf{z} * \mathbf{W}_1 + \mathbf{b}_1) * \mathbf{W}_2 + \mathbf{b}_2 \tag{3.8}$$

之后就是对隐层 $\mathrm{FFN}(\mathbf{z})$ 进行 dropout，最后加一个残差连接并层标准化。

3.3.2 解码器

解码器 (Decoder) 同样是由 6 个相同的 layers 层组成的，但是这里的 layer 和编码器是不一样的，包含了三个子层，多出了一个注意力子层。第一个子层是多头自注意力层，也是计算输入的 Self-Attention，但是因为这是一个生成过程，在时刻 i，大于 i 的时刻的目标端信息是看不到的，因此需要做信息遮蔽 (masking)，遮蔽的作用就是防止在模型在训练过程中使用未来会输出的单词。第二个子层是前馈全连接层，第三个子层是对编码器最后一层的输出 $\mathbf{so}_x$ 进行注意力计算的，从这里可以看出解码器的每一层都会对编码器的输出做 Multi Attention 的计算，这里的注意力函数就是普通的 Attention，而非 Self-Attention。

包含输入输出的解码过程如下:

* **输入**: 编码器的输出和对应 $i-1$ 位置解码器的输出。所以这里注意力函数不是 Self-Attention，它的键、值来自编码器的输出，查询则来自上一位置解码器的输出。

* **输出**: 在目标端第 i 位置的输出词的概率分布。

* **解码**: 在解码过程中要特别注意，编码可以并行计算，一次性全部编码出来，但解码过程不能并行地一次将所有序列解出来，而是像循环神经网络一样逐个进行解码，因为解码时需要上一个位置的输入作为注意力的查询。

3.3.3 位置信息编码

位置信息编码 (Positional Embedding) 在自然语言处理任务中是很重要的东西，可以观察到自注意力模型能很好地获得词与词之间的关联依赖关系，但是却不能得到词的位置信息，包括绝对位置和相对位置关系都无法提取。如果将注意力中的键和值的顺序打乱，得到的结果还是一样的。因此对位置信息编码来保留词序的信息是非常有必要的。Transformer 模型将每个词的位置进行编号，然后使用向量来表示对应的位置编号，最后将该位置向量和词向量相加得到新的词向量。模型使用不同频率的正弦和余弦函数来生成位置向量:

$$\begin{aligned} \mathbf{PE}_{pos,2l} &= \sin(\text{pos}/10000^{2l/d_{model}}) \\ \mathbf{PE}_{pos,2l+1} &= \cos(\text{pos}/10000^{2l/d_{model}}) \end{aligned} \tag{3.9}$$

pos 表示单词的位置，l 表示单词词向量的第 l 维，d_{model} 则是词向量维度大小。

位置向量不仅可以得到输入序列的绝对位置，还可以得到相对位置信息，根据公式:

$$\begin{aligned} \sin(\alpha+\beta) &= \sin(\alpha)\cos(\beta)+\cos(\alpha)\sin(\beta) \\ \cos(\alpha+\beta) &= \cos(\alpha)\cos(\beta)-\sin(\alpha)\sin(\beta) \end{aligned} \tag{3.10}$$

这表明在序列 $k+p$ 的位置向量可以看做是位置 k 的特征向量的线性映射，能够对模型得到序列之间的相对位置关系提供相当大的便利。

3.4 提高神经机器翻译鲁棒性方法

尽管语音识别是一个成熟的商业系统，但是它仍然存在非常多的识别错误。语音识别的识别错误可以被分为替换错误，删除错误和插入错误，我们在表 Table 3.2说明了错误类型占比。

我们在内部数据计算了这三种错误的词错误率 (WER)，数据来源于 100 个小时的中文语音数据，从表中可以看出替换错误是主要的错误类型，”Others” 包

错误类型	错误率	
Ground Truth	-	语 音 翻 译. yǔ yīn fān yì.
Substitution	9.4%	语 音 翻 一. yǔ yīn fān yī.
Deletion	2.3%	音 翻 译. yīn fān yì.
Insertion	0.7%	语 音 翻 了. yǔ yīn fān le .
Others	16.7%	

表 3.2 三种语音识别错误的识别错误率

Table 3.2 Error rates of three error categories for ASR

含了一些复杂的错误类型，包括说话人口音和嘈杂的环境等所导致的错误。另一些研究也已经证明，超过 50% 的机器翻译错误与替换错误有关，替代错误比删除或插入错误对翻译质量的影响更大 (Vilar 等, 2006; Ruiz 和 Federico, 2014)。替换错误可进一步分为两类：同音词之间的替换，定义为同音异形字错误（Similar Pronunciation，SP errors) 和发音相似的词之间的替换，定义为相近音异形字错误（Homophone Words，HM errors)，在这些结论的基础上，本文重点研究了这两种替代错误。在接下来的章节中，我们将以汉语为例介绍我们的方法，我们的方法同样可以以类似的方式应用到其他语言中。

我们的方法旨在提高神经机器翻译对语音识别错误的鲁棒性。为此首先构造一个与测试数据具有相似分布的训练数据集，然后利用语音信息来将相近音异形字错误 (SP errors) 和同音异形字错误 (HM errors) 纠正。具体地说，我们的方法步骤如下：

1. 在训练集的句子中随机添加相近音异形字错误和同音异形字错误，来模拟 ASR errors；

2. 预测含有相近音异形字错误汉字的正确的读音，并作出修改；

3. 将正确的语音信息集成到含有同音异形字错误的汉字隐层信息中，来改进同音异形字错误。

图 3.3展示了我们方法的架构，红色标记表示输入或词向量可能含有噪音，蓝色

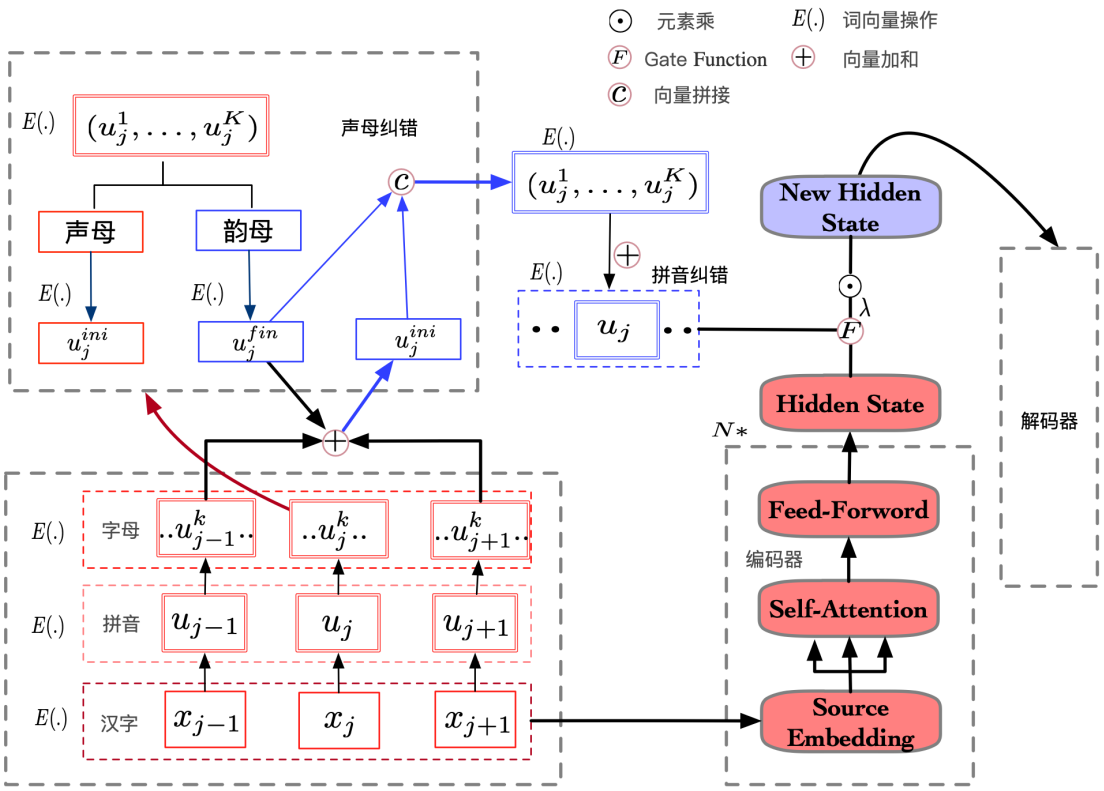

图 3.3 噪音模型的编码器架构图

Figure 3.3 Encoder architecture of our noise model

表示纠正过后的词向量，这两种错误的解决是依次进行的，我们首先通过韵母和上下文拼音来预测正确的声母，从而得到正确的拼音，这是解决 SP errors 的关键步骤，然后使用纠正后的拼音信息通过一个 gate network 再对可能含有噪音的汉字进行纠正，这是同时解决 SP errors 和 HM errors 的关键。我们将在接下来的章节中详细介绍我们的模型。

3.4.1 训练集模拟语音识别错误

首先对每个源端字进行处理，以一定的概率 $p \in [0, 1]$ 来决定是否将该字替换为语音识别噪音字，如果被选中，则挑选一个与同音异形字或相近音异形字来该字进行替换，这个噪音字的选择也是按照字在训练集中出现的频率来选择的给定一个源端字 x，有相近音异形字集 $\mathcal{V}_{sp}(x)$ 和同音异形字集 $\mathcal{V}_{hm}(x)$，$\mathcal{V}_{sp}(x)$ 和 $\mathcal{V}_{hm}(x)$ 通过对大规模训练集进行统计得到，按照概率 p 的伯努利分布来进行采样替换：

$$r_x \sim \text{Bernoulli}(p) \tag{3.11}$$

$r_x \in \{0,1\}$ 是伯努利分布的输出，$p \in [0,1]$ 是伯努利分布的输出为 1 的概率。当 r_x 是 1 的时候，从噪音集 $\mathcal{V}(x)$ 挑选一个当前字 x 的噪音字 x_* 来代替当前词，某个噪音字 x_* 被选中的概率为：

$$p(x_*) = \frac{\text{Count}(x_*)}{\sum\limits_{x'_* \in \mathcal{V}(x)} \text{Count}(x'_*)} \tag{3.12}$$

$\text{Count}(x_*)$ 表示字 x_* 在训练集出现的次数，为了得到和语音识别分布类似的训练数据，我们在构造的噪音集 $\mathcal{V}(x) = \mathcal{V}_{sp}(x) \cup \mathcal{V}_{hm}(x)$ 采样得到。

3.4.2 相近音异形字错误修正

对于中文来说，拼音是用来表示汉字的发音，每个汉字的拼音都包含几个字母。根据发音规则，一个拼音可以被分成两部分，声母和韵母。声母通常是拼音的第一个字母，韵母则是其余的字母集合。我们通过观察语音识别的识别结果，发现大部分相近音异形字的错误是因为声母的识别错误，而韵母是正确的。另外，可以看到汉语拼音是有由声母 + 韵母的固定搭配。基于此，可以采用忽略声母，使用韵母预测声母的方式来对可能出现错误的声母进行纠正，从而达到纠正整个拼音特征的目的。值得说明的是我们预测正确的声母，其实是预测每个声母在声母表中的概率分布，这让我们可以利用神经网络来进行训练。

给定一个源端句子 $\mathbf{x} = (x_1, \ldots, x_J)$，用 $\mathbf{u} = (u_1, \ldots, u_J)$ 来表示它对应的拼音序列。用 u_j^k 表示拼音 u_j 的第 k 个字母。对于拼音 u_j，它的声母可以表示为：

$$u_j^{\text{ini}} = u_j^1 \tag{3.13}$$

另外，它的韵母可以表示为：

$$u_j^{\text{fin}} = [u_j^2, \ldots, u_j^K] \tag{3.14}$$

其中，K 是拼音 u_j 的字母个数。

我们同样拥有两份词表，包括拼音词表和拼音字母表，因此对应的也有两个不同词向量矩阵。可以通过加和的形式得到韵母 u_j^{fin} 的词向量：

$$\mathbf{E}[u_j^{\text{fin}}] = \sum_{k=2}^{K} (\mathbf{E}[u_j^k]) \tag{3.15}$$

$\mathbf{E}[\cdot]$ 表示某个输入的词向量表示，对于相近音异形字的错误，之前提到是由于声母的错误导致的，可以通过联合该拼音的韵母词向量和该拼音的上下文词向量(即前一个词对应的拼音 u_{j-1} 和后一个拼音 u_{j+1})，来预测正确的声母的概率分布。对于 u_j 可以得到它的声母概率分布：

$$p^{\text{ini}} \sim \text{softmax}(g^{\text{ini}}(\mathbf{E}[u_{j-1}] + \mathbf{E}[u_j^{\text{fin}}] + \mathbf{E}[u_{j+1}])) \tag{3.16}$$

$g^{\text{ini}}(\cdot)$ 是线性映射函数，使用加权求和的形式来对所有可能的声母的词向量通过预测的概率进行加和，来得到正确的声母词向量 u_j^{ini}：

$$\mathbf{E}[u_j^{\text{ini}}] = \sum_{l \in \mathcal{V}^{\text{ini}}(u_j)} p^{\text{ini}}(l) * \mathbf{E}[l] \tag{3.17}$$

$\mathcal{V}^{\text{ini}}(u_j)$ 表示汉语拼音字母表，$p^{\text{ini}}(l)$ 表示在公式 Eq 3.16预测的字母概率。最终更新拼音 u_j 的词向量为：

$$\mathbf{E}[u_j] = g(\mathbf{E}[u_j], \mathbf{E}[u_j^{\text{ini}}], \mathbf{E}[u_j^{\text{fin}}]) \tag{3.18}$$

$g(.)$ 是一个线性映射函数。

3.4.3 同音异形字错误修正

对于同音异形字错误来说，可能是汉字是错误的但是汉字对应的拼音是正确的，所以处理方式要和相近音异形字有很大区别，同音异形字的拼音可以提供正确的信息来直接对源端汉字进行改正。根据这一个特点将拼音的词向量来对编码器的隐层输出序列 $(\mathbf{h}_1, \ldots, \mathbf{h}_J)$ 来进行修正，以此来得到更准确地源端隐层表示。

我们可以通过一个 gate network 来对源端汉字的隐层表示进行修正，之所以采用 gate network 是希望能在正确的拼音信息和当前可能存在噪音的字的隐层信息中做个取舍，比例可以由参数训练得到，输入的汉字是正确的，在模型中汉字的权重比例会变大，当汉字是噪音字时，对应的拼音的权重比例会变大，这样既可以有效的利用到正确的拼音信息也可以避免当前字的翻译出现大的偏差。计算第 j 个源端字与拼音的 gate 权重 λ_j 值：

$$\lambda_j = \mathbf{W}_\lambda \tanh\left(\mathbf{W}_h \mathbf{h}_j + \mathbf{W}_u \mathbf{E}[u_j]\right) \tag{3.19}$$

$\mathbf{W}_\lambda$, $\mathbf{W}_h$ 和 $\mathbf{W}_u$ 都是参数矩阵，通过 λ_j 我们可以动态的调节汉字和拼音所占比例，以此来更新源端隐层表示：

$$\mathbf{h}_j = \lambda_j * \mathbf{h}_j + (1 - \lambda_j) * \mathbf{E}[u_j] \tag{3.20}$$

最终，更新后的源端隐层表示被用于解码器，计算注意力向量和预测目标词。

3.5 实验

3.5.1 数据准备

本章主要是为了解决中文环境下的噪音输入问题，因此数据集采用了中英数据，主要包括两个数据集：

NIST Zh→En: 中英数据集中包括 1.25M 双语语对[1]. 我们选择 NIST 2002 作为我们的验证集，包括 878 句双语对，选择 NIST 2003, 2004, 2005, 2006 作为我们的**干净**测试集，分别包括 919, 1788, 1082，1664 句双语对，**干净**测试集中不包含语音识别错误。

CWMT17 ZH→EN: 另外我们还准备了更大数据的训练集，来验证我们模型的有效性，CWMT 的中英数据共包含 9.3M 的双语对，使用 newsdev2017 和 newstest2017 分别作为验证集和**干净**测试集。

对于这两个数据集，使用 Moses scripts[2]来对英文端做了 tokenized。并且字词分割使用了 Byte-Pair Encoding(BPE) (Sennrich 等, 2016)，BPE 的合并次数设置为 30k。对于中文端，将句子分割为中文字。使用 ChineseTone[3]工具来获得汉字对应的拼音。

我们在 section **??**提到如何在干净的测试集中模拟相近音异形字的噪音，同音异形字噪音和这两种结合的噪音。我们将替换概率 p 分别设置为 0.1，0.2，03，从而模拟不同程度的噪音环境。在干净的测试集上添加上述的三种噪音来验证我们模型的鲁棒性，这样能够得到与干净测试集相对应的三个噪音测试集。

为了进一步验证系统对于真实环境下的噪音输入的鲁棒性，我们对 CCMT 2019 语音翻译任务中的测试集进行评估，该测试集来自真实的 ASR 识别结果，测试集共包含 956 句双语对。可以从这个链接[4]进行下载。

[1]双语对主要是从以下的数据集中抽取的 LDC2002E18, LDC2003E07, LDC2003E14, Hansards portion of LDC2004T07, LDC2004T08 and LDC2005T06

[2]http://www.statmt.org/moses/

[3]https://github.com/letiantian/ChineseTone

[4]http://ai.baidu.com/broad/download

3.5.2 训练细节

基线系统是基于 Transformer 模型 (Vaswani 等, 2017b) 的，Transformer 模型在之前的章节有说明，这里采用的系统是开源工具 Fairseq-py (Edunov 等, 2017)。系统的基础参数完全依照 Vaswani 等 (2017b) 论文中提到的 Base 模型的参数来设置的。所有模型的训练都是在一个 8 个 NVIDIA TITAN Xp GPUs 卡的单服务器上训练的，batch size 设置为 4096 tokens，句长最长为 100。对于基线系统，训练 Step 达到 100k，每隔 1000 个 Step 保存一次模型参数，最后对最近的 5 个 Checkpoints 进行参数平均之后用测试集进行验证。

在解码过程中，设置 beam size 为 5，length penalty α=0.6 (Wu 等, 2016)。另外对所有系统使用 case-sensitive NIST BLEU (Papineni 等, 2002) scores，最后的 BLEU 评价使用 *multi-bleu.pl*。

3.5.3 实验结果

System	p	NIST Clean	NIST Artificial Noise				CCMT Speech
			1 Sub	2 Subs	3 Subs	Ave.	
Baseline	-	45.21	43.63	42.24	41.33	42.40	11.08
Baseline+Pinyin	-	45.11	43.88	42.83	42.02	42.91	
Our Method	0.1	45.15	44.64	44.23	43.87	44.24	-
	0.2	45.13	44.83	44.41	44.12	44.45	11.45
	0.3	44.95	44.68	44.45	44.09	44.40	-

表 3.3 NIST 数据集上的噪音实验结果

Table 3.3 Results on NIST test datasets

实验的结果我们在表 Table 3.3可以看到，结果是测试集 nist03，nist04，nist05，nist06 上的 BLEU 平均值，测试集包括干净测试集和三个噪音测试集，噪音测试集中的 1 Sub, 2 Subs 和 3 Subs 表示该测试集句子分别包括 1，2，3 个噪音，噪音是 HM errors 和 SP errors，p 是替换率。Baseline+Pinyin 系统是在基线模型的基础上，只利用拼音来对汉字的隐层表示进行修改，汉字在这个系统的输入是干净不含噪音的。在实验结果表明，无论是在 NIST 数据集任务还是 CWMT 数据集任务上，我们提出的模型相比于基线在噪音测试集上有了巨大的提升。另外，还可以得出以下结论：

首先，基线系统在干净的测试集上表现的相当优异，但是在噪音测试集上性能下降非常明显，这说明当前先进的神经机器翻译系统在真实噪音环境下的表现是不能让人满意的，这在之前的工作 Belinkov 和 Bisk (2018); Cheng 等 (2018) 也有被提到。

另外，我们的模型在干净的测试集上同样表现优异，相比于基线系统并没有性能的下降，并且在噪音测试集上结果仅仅只比干净测试集稍差一点，比基线系统在噪音测试集上的结果要好非常多，在 CWMT 大数据集上的结果要比基线系统高 2.44 BLEU。这说明我们系统的鲁棒性更强，针对语音识别错误表现出了很强的拟合能力。

我们方法在超参 p 为 0.2 的时候取得了最好的结果，这说明不同程度的噪音采用对最终结果的影响也是比较大的，在实际训练中，太多或太少的模拟语音识别噪音采样都不能取得最好的结果。这个结论能够很好的指导我们如何训练一个更鲁棒的系统。而且当我们不对汉字输入进行替换，只利用拼音进行重写，结果相对于基线系统仍然有 0.51 BLEU 的提高，进一步证明拼音这一特征的有效性，但相对于我们的模型，BLEU 还有较大差距，表明不同程度噪音的模拟会让拼音特征的作用发挥到最大。

最后可以看到，我们的模型在来自 CCMT2019 的真实场景下的测试集上的结果，依然相对于 baseline 有 0.37 BLEU 的提。相对于在我们模拟的测试集上的实验结果，提升相对较小，可以在表3.2看到，一些不可统计错误“Others”占比较大，而在 CCMT2019 的测试集上，环境更加嘈杂，导致测试集中“Others”的占比更大一些，而替换错误相对占比更小，这是导致我们在真实环境的测试集提升没有模拟的测试集明显的原因。我们的系统在出现替换错误的句子中依然保持了较强的鲁棒性。

System	p	Clean	Noise Ave.
Baseline	-	45.21	42.40
+SP Amendment	0.2	45.20	43.55
+HM Amendment	0.2	45.30	43.77
+Both Amendment	0.2	45.13	44.45

表 3.4 NIST 数据集上的消融实验结果

Table 3.4 Ablation results on NIST test datasets

System	p	Clean	Noise
Baseline	-	23.11	20.23
+SP Amendment	0.2	23.08	22.12
+HM Amendment	0.2	23.09	22.23
+Both Amendment	0.2	23.13	22.67

表 3.5 CWMT17 数据集上的消融实验结果

Table 3.5 Ablation results on CWMT test datasets

3.5.4 消融实验

为了进一步说明我们模型各个部分的作用，单独对各个模块进行了测试。首先就是只针对相近音异形字的错误，再对同音异形字的错误进行测试，最终将两个模块进行结合来测试最终的结果。总的结果在表 Table 3.4(”+SP Amendment”, ”+HM Amendment” 和”+Both Amendment” 分别表示模型只采用相近音异形字的错误，只采用同音异形字错误，和同时采用两种错误的情况。) 和表 Table 3.5进行的了说明。”+SP Amendment” 的方法提高了系统的鲁棒性和对错误的拟合能力，很明显在所有的情况下，我们的模型在两个数据集上要比基线系统高 +1.15 和 + 1.89 BLEU。”+HM Amendment” 的方法进一步提升系统在噪音测试集上的表现，在 NIST 任务和 CWMT 任务上分别取得了 +1.37 和 +2.00 BLEU 的提升。这说明拼音信息在中文输入条件下是一个非常重要且有效的特征，尤其是语音环境下。

最后，在同时使用相近音异形字和同音异形字两个特征的情况下，系统获得了最佳性能，这表明这两个特征可以相互融合，能够进一步提高模型鲁棒性。

3.5.5 Training Cost

我们的实验也对 Training Cost 做了对比，比较我们的模型和基线系统在训练时的 cost 变化。可以从图.3.4看到整个 cost 趋势。

从 Training Cost 的趋势图可以看到，我们模型的训练代价相比于基线系统稍高，这表明我们提出的模型在预测下一个单词时可以考虑更多的词，因为它综合了源端字符的拼音信息。这在测试集上得到比基线系统更高的 BLEU 值，会忽略一些没有发音信息的更合适的单词候选者。Training Cost 曲线和测试集上的 BLEU 结果表明，该方法有效地改善了基于干净训练数据的神经网络模型的泛化

图 3.4 Training Cost 对比图

Figure 3.4 Training cost of the baseline model and our robust system

性能和抗噪能力。

3.5.6 句长测试

在本章中，评估了我们模型在不同长度的源端句子下的表现，结果可以在图.3.5看到，可以很明显的看出无论是我们的模型还是基线系统，在句长 50 之前性能是一直提升的，但当句长超过 50 之后，翻译质量就有了较大的下降，这个结论在之前的论文中也有提到 (Bahdanau 等, 2015)。

此外，句长曲线也表明长句子可以提供更多的上下文信息和更丰富的语音信息，这对噪声消歧是有帮助的。在每个长度间隔内，我们系统在所有噪音测试集上的性能都优于基线模型的。源句中噪音数量的增加并没有太大程度地降低我们模型的性能，说明了该方法对于噪音输入具有很强的鲁棒性。

3.5.7 示例说明

在表 Table 3.6，我们提供了两个实际环境下的语音翻译的例子，包括同音异形字替换错误和相近音异形字替换错误，来说明我们模型对于语音识别输入的鲁棒性。在这两个示例中，原句子的句法结构和语音信息被噪音字破坏，在语音环境中，正确的“触发”被语音识别系统识别成“处罚”，这两个字的发音是相同的，属于同音异形字替换错误，正确的“同仁”被语音识别系统识别成“红人”，属于相近音异形字替换错误。人类对于这种错误是可以明白其正确的含义的，但

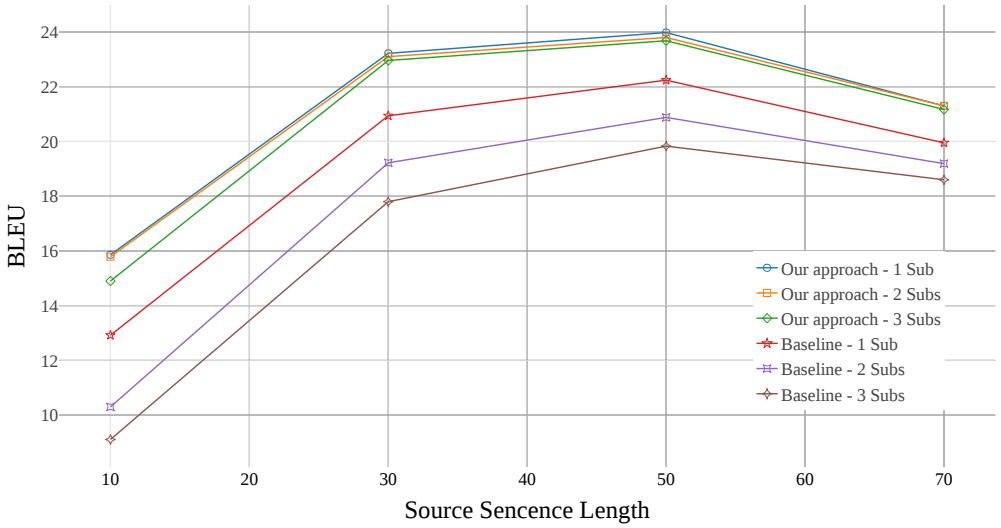

图 3.5 在不同源端句子长度的实验结果

Figure 3.5 Effect of source sentence lengths of input

是对于机器翻译模型来说，“处罚”和“红人”是两个高频词，是具有丰富语义的词，神经机器翻译是无法翻译出正确的译文，从基线系统的输出结果也可以看出，翻译是错误的。与之相对的是我们模型的译文，将“触发”和“同仁”正确的翻译出来。这个示例结果表明，我们系统具有较强的语音识别噪音纠错能力，系统鲁棒性的提高主要归功于我们提出的语音识别模拟噪声的训练和汉语拼音特征的加入。

Gold input	动作	的	**触发**	和	构成	顺序	是	可以	调整的
	dòng zuò	de	**chù fá**	hé	gòu chéng	shùn xù	shì	kě yǐ	tiáo zhěng de
ASR-HM	动作	的	**处罚**	和	构成	顺序	是	可以	调整的
	dòng zuò	de	**chǔ fá**	hé	gòu chéng	shùn xù	shì	kě yǐ	tiáo zhěng de
Baseline	The order of punishment and composition of actions can be adjusted								
Our Approach	The trigger and composition of the action can be adjusted								
Gold input	我想	这个	是	我们	很多	业界	**同仁**	反思	的地方
	wǒ xiǎng	zhè gè	shì	wǒ men	hěn duō	yè jiè	**tóng rén**	fǎn sī	de dì fāng
ASR-SP	我想	这个	是	我们	很多	业界	**红人**	反思	的地方
	wǒ xiǎng	zhè gè	shì	wǒ men	hěn duō	yè jiè	**hóng rén**	fǎn sī	de dì fāng
Baseline	I think this is a place for a lot of our industry red people to reflect on								
Our Approach	I think this is a place where a lot of us in the industry are reflective								

表 3.6 我们模型与基线系统模型在真实环境下的对比示例

Table 3.6 An example of our model and baseline system model

3.6 本章小结

近几年来，语音输入变得越来越流行，因此先进的机器翻译系统必须处能够理包含 ASR 语音识别错误的输入。在本文中，我们从两个方面来提高神经机器翻译在输入中包含语音识别错误时的鲁棒性。一种是从数据的角度出发，在训练数据中加入模拟的语音识别噪音，使训练数据和测试数据具有一致的分布。另一种是从模型本身的角度出发，我们的方法能够处理两种最广泛存在的语音识别错误: 相近音异形字之间的替换错误 (SP errors) 和同音异形字之间的替换错误 (HM errors)。对于 SP errors，我们利用上下文的语音信息来纠正拼音单词的词向量。对于 HM errors，我们直接利用正确的语音信息来修改源词的隐层表示。实验结果证明了我们模型的有效性，我们的方法能够很好地处理这两种类型的噪音，对错误的输入具有较强的鲁棒性。

第 4 章　基于拼音信息的篇章级机器翻译对于噪音输入纠错的研究

4.1　介绍

当前流行的神经机器翻译在不考虑篇章上下文的情况下，对整个模型进行训练，在一些公开测试集上的结果来看，翻译模型已经在各种翻译任务中取得了非常好的结果。但是在日常生活中的翻译，用户的输入往往并不是独立的一句话，而是像文档一样，是一系列连贯的输入。但是翻译系统由于忽略当前输入句子一些有价值的语境信息，无法建立当前句子与上下文的语义联系，这种信息的忽略可能会降低翻译时的连贯性和忠实度，以至于翻译结果出现偏差。篇章信息能够提供丰富的指代信息，对于一些歧义词，能够起到很好的辅助翻译的作用。即使基于 Transformer 模型的神经机器翻译，在忽略篇章的上下文的情况下，对于篇章翻译的表现也不如人意。近年来利用篇章信息来增强神经机器翻译的性能也成为一个热点的研究方向，篇章翻译一般是用于解决一致性问题 (包括指代，时态等) 和歧义问题。

篇章上文	**(1)**	一会儿我会放弃掉所有**视频**的播放，以节省时间呐;
篇章上文	**(2)**	去年这个系统在乌镇世界互联网大会上就首秀过;
当前输入	–	这是第二次，那上一次的时候还是只是给了**字母**。

表 4.1 源端篇章信息示例

Table 4.1 Example of document-level content

由于语音识别在生活中的广泛使用，现在机器翻译的输入很多情况下是含有篇章结构的语音输入，比如像演讲的同声传译。而和正常篇章翻译不同的是，语音翻译的输入往往是含有大量的识别错误的，当前最先进的 Transformer 翻译系统对噪音输入的表现也不能让人满意。我们受篇章翻译的启发，利用语音输入的上文信息来解决噪音输入的问题，通过之前识别内容提供的上文信息来对当前噪音输入进行纠正。对于篇章翻译下的噪音问题，可以从例子4.1看到，通过篇章上文中的“视频”(红色标记)，可以得到当前句子中的“字母”(蓝色标记)

是错误的，应该是“字幕”。篇章信息对于解决噪音问题可以提供比较有参考价值的信息。

传统篇章翻译的做法也是包括两点，一个是如何从前几句提取出需要的信息，二是如何利用提取的信息来对当前词进行改进和信息补充。我们针对篇章翻译的特点和真实环境下的噪音问题，提出了基于篇章信息的机器翻译对于语音识别输入的鲁棒性改进方法，同时在拼音与汉字之间构建了一个统一化的语义空间来提取篇章上文中的有效信息，最后对可能含有噪音的源端输入序列的隐层表示进行改写。我们的模型在 Ted 的测试语料上取得较高的改进，表明了我们方法的有效性。

4.2 相关工作

篇章翻译一直是机器翻译领域的一个重要研究方向，在统计机器翻译 (SMT) 时代就已经有一些工作来尝试将篇章信息引入到机器翻译，(Hardmeier 和 Federico, 2010) 使用词依赖模型来识别源文档中词与词之间的联系，以改进代词的翻译。(Gong 等, 2011) 使用基于缓存机制的模型从预先生成的翻译中提取相关联的信息，并将其用于增强篇章翻译。(Garcia 等, 2014) 提出了一种两阶办法，来对普通句子翻译模型进行改进。

在当前比较流行的神经机器翻译，篇章翻译也愈发引起研究人员的重视，(Jean 等, 2017) 提出方法，通过注意力机制来对前一句的单词信息进行提取，来对当前解码器进行调节，提高传统神经机器翻译模型的性能。(Wang 等, 2017) 使用一个两级的层级 RNN 来对之前的三个源句子进行编码，然后将其用作计算解码器隐藏状态的附加输入。(Bawden 等, 2018) 使用多编码器模型来对篇章目标端上文信息进行编码，他们强调了目标端上下文的重要性。

Voita 等 (2018) 也利用篇章信息来对 Transformer 系统进行改进，但我们的方法的目的和在建模训练时的策略有所不同，实验部分也有很大的不同。我们希望篇章信息能够提供对于噪音输入起到纠正作用的信息，主要是提高神经机器翻译的鲁棒性。

关于文档级的大多数现有工作都侧重于将篇章级的上下文纳入机器翻译模型中。这些方法大致可以分为两大类：一是计算整个文档级上下文的隐层表示，并用隐层表示信息来辅助机器翻译计算，二是使用 Cache 缓存机制来存储篇章上下文中与源端输入最相关的信息。我们的方法属于第一类，使用多头注意力机制来表示和集成篇章信息。

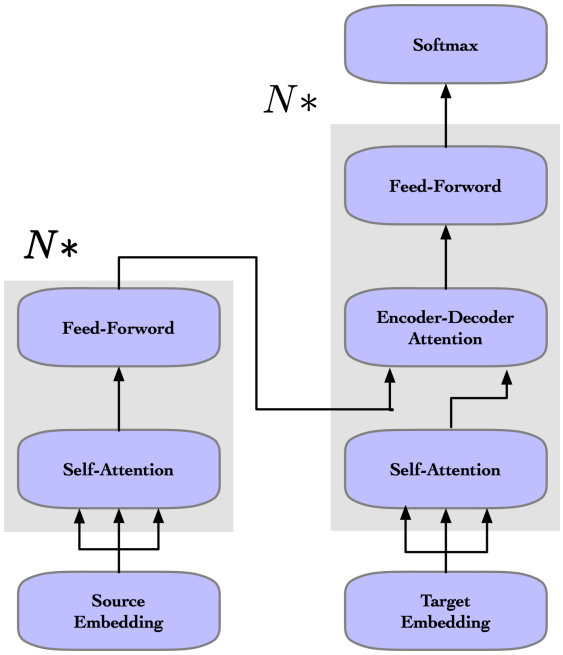

图 4.1 Transformer 模型的抽象化架构

Figure 4.1 The Transformer model architecture

4.3 系统背景

本章针对当前最先进的 Transformer 系统进行改进，在篇章上文提供了非常有用的信息之后，使用我们的模型能够进一步提高模型对于噪音输入的鲁棒性。

由于篇章信息通常包含几个句子，因此捕获篇章句子之间的长距离依赖关系 (long-range dependencies) 并得到与当前源句相关联的信息是非常重要的。使用多头注意力机制 (multi-head Self-Attention)(Vaswani 等, 2017b) 来计算篇章上文的隐层表示，因为它能够将长序列之间的最大路径长度减少到 $O(1)$，并确定在篇章句子中不同位置的单词对于当前句子的重要程度 (Bahdanau 等, 2015)。多头注意力机制的这一特征也被证明在其他自然语言处理任务中也是有效的。

我们可以从图.4.1看到 Transformer 模型的基础架构，如图所示，Transformer 模型的编码器每一层包括两个子层，解码器每层包括三个子层，具体关于 Transformer 系统的描述，在第三章中有详细介绍，在本章就不再具体介绍。我们将整个编码器的过程用一个函数 Encoder(.) 来定义，编码器函数 Encoder(.) 的输入是输入序列，可以是汉字序列或拼音序列，编码器的输出是输入序列对应的隐层表示序列，是整个输入的向量表示，能够表达当前输入的上下文信息。

篇章上文 (1)	..	放	弃	掉	所	有	**视**	**频**	的	播	..
	..	fàng	qì	diào	suǒ	yǒu	**shì**	**pín**	de	bō	..
篇章上文 (2)	..	乌	镇	世	界	互	联	网	大	会	..
	..	wū	zhèn	shì	jiè	hù	lián	wǎng	dà	huì	..
当前输入	..	时	候	是	只	是	给	了	**字**	**母**	..
	..	shí	hou	hái	zhǐ	shì	gěi	le	**zì**	**mǔ**	..

表 4.2 篇章信息解决噪音问题示例

Table 4.2 An example with noise of document-level content

4.3.1 拼音特征

拼音作为输入的语音特征，在语音翻译尤其是同传环境下有这非常重要的作用，语音翻译下的输入噪音往往是拼音的识别错误，像同音异形字的噪音错误，篇章信息中蕴含非常丰富的语音信息，相比于识别的中文汉字，汉字所对应的拼音特征要准确的多, 我们通过表4.2可以看到，红色标记的“字母”是语音识别的噪音，正确的应该是“字幕”，汉字是错误的，但是对应的拼音“zimu”是正确的。如何正确利用这个正确的拼音，并与篇章句子建立联系，是正确纠正噪音的关键。

4.4 篇章信息解决噪音输入问题

首先在训练集中模拟噪音输入，来保持训练时我们的模型能对噪音输入具有鲁棒性，然后使用多头注意力机制来将篇章信息集成到神经机器翻译模型中的编码器, 并探讨如何通过拼音特征建立篇章信息与当前源端句之间的联系。通过双层的注意力函数来解决噪音输入的问题，主要包括两部分：

* 首层注意力函数得到各特征输入的隐层表示，这里的注意力函数是多头自注意力，实际做法是多个编码器来得到输入序列的隐层表示，输入序列包括当前源端输入、对应的拼音序列和篇章上文。

* 第二层注意力函数来提取篇章上文中与当前输入相关联的关键信息，并通过一个 gate network 来将篇章信息引入到神经机器翻译模型中去

我们模型的架构在图.4.2可以看到，接下来将对我们提出的方法进行具体阐述。

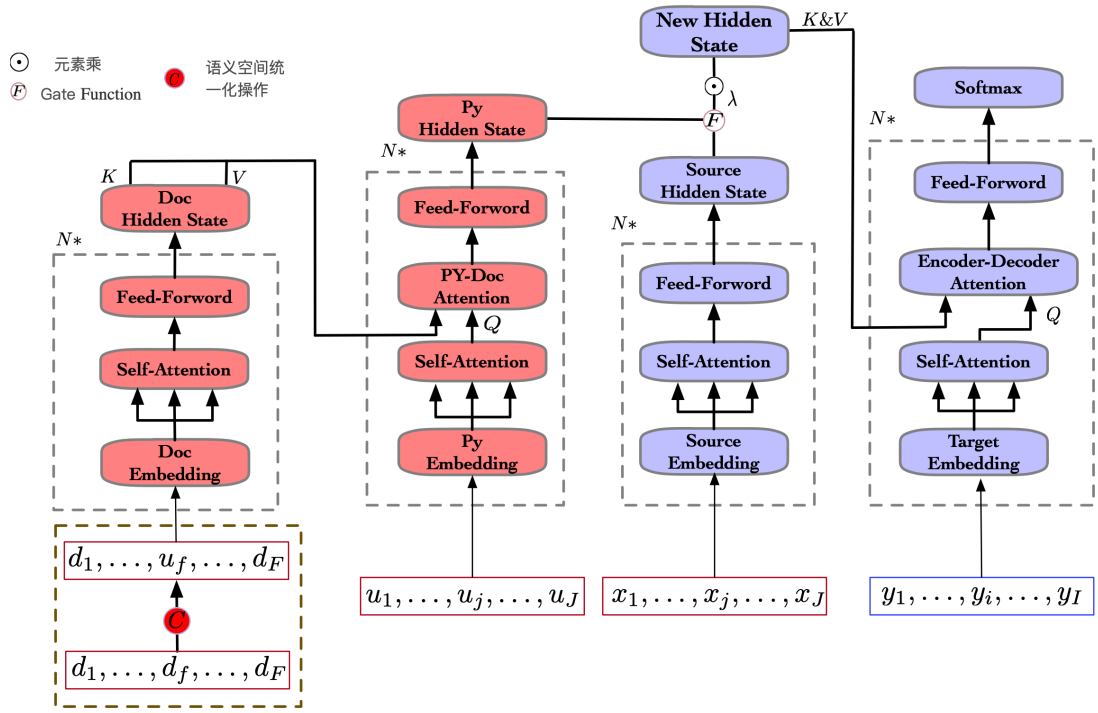

图 4.2 篇章噪音模型架构图

Figure 4.2 Our document-level model architecture

4.4.1 训练集模拟噪音输入

首先对每个源端的正常输入 (并不包括篇章句子) 中的字进行处理，以一定的概率 $p \in [0,1]$ 来决定是否将该字替换为语音识别噪音字，如果被选中，则挑选一个与同音异形字来该字进行替换，这个同音异形字的选择也是按照字在训练集中出现的频率来选择的。给定一个源端字 x，有同音异形字集 $\mathcal{V}_{hm}(x)$，按照概率 p 的伯努利分布来进行采样替换：

$$r_x \sim \text{Bernoulli}(p) \tag{4.1}$$

$r_x \in \{0,1\}$ 是伯努利分布的输出，$p \in [0,1]$ 是伯努利分布的输出为 1 的概率。当 r_x 是 1 的时候，从噪音集 $\mathcal{V}(x)$ 挑选一个当前字 x 的噪音字 x_* 来代替当前词，某个噪音字 x_* 被选中的概率为：

$$p(x_*) = \frac{\text{Count}(x_*)}{\sum_{x'_* \in \mathcal{V}(x)} \text{Count}(x'_*)} \tag{4.2}$$

$\text{Count}(x_*)$ 表示字 x_* 在训练集出现的次数，为了得到和语音识别分布类似的训练数据，在构造的噪音集 $\mathcal{V}(x) = \mathcal{V}_{hm}(x)$ 采样得到。

4.4.2 统一化语义空间下的篇章信息提取

在本章中,给定一个当前源端句子 $\mathbf{x} = (x_1^t, \ldots, x_j^t, \ldots, x_J^t)$,使用 $\mathbf{u} = (u_1^t, \ldots, u_j^t, \ldots, u_J^t)$ 来表示它对应的拼音序列，长度与源端句子长度相同,其中 t 表示当前句子是文档中的第 t 句。当前源端句子对应的篇章长句是与当前原句最接近的三句，用 $\mathbf{d} = (d_1^{t-3}, \ldots, d_A^{t-3}; d_1^{t-2}, \ldots, d_B^{t-2}; d_1^{t-1}, \ldots, d_E^{t-1})$ 来表示篇章长句，A，B，E 表示篇章三句中每句长度，篇章长句的总长度为 $F = A + B + E$。在实际中，并没有独立区分篇章上文的三句，而是将篇章三句按照顺序拼成一句长句，因为拼接成的篇章长句中，篇章长句的顺序关系可以提供绝对位置信息，每句话之间的句号分割可以表明新的子句的开始，可以提供相对位置信息。所以将篇章长句的重新表示为 $\mathbf{d} = (d_1, \ldots, d_f, \ldots, d_F)$。

通过上文的介绍，针对噪音的源端输入，我们希望通过源端句子对应的拼音，来建立篇章中的某些有用信息与源端输入的关系。在本章节提出了统一化的语义空间来将汉字与拼音的向量表示统一。在表4.2中可以看到“字母”和“视频”在语义上是不相关的，而“zimu”与“视频”是相关的(因为字幕这个词)，这在逻辑上和人类经验上是成立的，为了让计算机也学会“zimu”与“视频”的关系，采用一个统一化的语义空间，相当于一个共享词表，在训练过程中，一句话不仅仅是汉字的集合，而是汉字和拼音的集合，比如，“那上一次的时候还是只是给了字母”在训练时候就可能变成“na 上一次的时候 haishi 只是给了 zimu”。而目标端一直是不变的，其实这在训练的时候就已经强制的约束，“zimu”和“字幕”是强相关，同时“字幕”与“视频”是强相关，从而得到“zimu”与视频的强相关关系。

这里传统的篇章输入 $\mathbf{d} = (d_1, \ldots, d_f, \ldots, d_F)$ 便被改写为 $\mathbf{d} = (d_1, \ldots, d_{f_1}, \ldots, u_{f_2}, \ldots, d_F)$，由全部是汉字的序列变为汉字与拼音混合的序列，$f_1$ 和 f_2 分别表示篇章长句中第 f_1 和 f_2 位置的词，第 f_2 个词被替换成了拼音。篇章输入的哪些位置的汉字应该被拼音替换，使用采样来选中替换位置，按照概率 $p = 0.2$ 的伯努利分布来进行采样替换,具体采样方法可以参看章节4.4.1，值得一提的是这里这涉及采样后的拼音替换，和同音异形字和相近音异形字并无关系。最终通过编码器函数 Encoder(.) 可以得到三个不同输入的编码器输出:

$$\mathbf{H}^x = \text{Encoder}(x_1^t, \ldots, x_j^t, \ldots, x_J^t) \tag{4.3}$$

$$\mathbf{H}^u = \text{Encoder}(u_1^t, \ldots, u_j^t, \ldots, u_J^t) \tag{4.4}$$

$$\mathbf{H}^d = \text{Encoder}(d_1, \ldots, d_{f_1}, \ldots, u_{f_2}, \ldots, d_F) \tag{4.5}$$

$\mathbf{H}^x = (\mathbf{h}_1^x, \ldots, \mathbf{h}_j^x, \ldots, \mathbf{h}_J^x)$ 表示当前源端字输入的隐层表示序列，$\mathbf{H}^u = (\mathbf{h}_1^u, \ldots, \mathbf{h}_j^u, \ldots, \mathbf{h}_J^u)$ 表示当前源端拼音输入的隐层表示序列，$\mathbf{H}^d = (\mathbf{h}_1^d, \ldots, \mathbf{h}_{f_1}^d, \ldots, \mathbf{h}_{f_2}^d, \ldots, \mathbf{h}_F^d)$ 表示篇章上文输入的隐层表示序列。

在上文提到，如何通过拼音建立当前源端字与篇章中关键信息的联系，我们使用一个 Attention 函数来提取篇章中的信息，称之为 Doc_Attention，具体如下：

$$\mathbf{c}_j^d = \text{Attention}(\mathbf{h}_j^u, \mathbf{H}^d, \mathbf{H}^d) \tag{4.6}$$

$\mathbf{c}_j^d$ 是当前输入的第 j 拼音所对应的篇章信息的注意力向量。Doc_Attention 的 Query 是当前第 j 源端拼音的隐层表示 $\mathbf{h}_j^u$；Key 和 Value 相同，是篇章上文的隐层表示序列 $\mathbf{H}^d$。通过统一化的语义空间来建立不相互对应的拼音与汉字之间的联系，通过 Doc_Attention，以拼音作为 Query，我们模型能够提取到篇章上文中最有效最关键的信息。

4.4.3 篇章信息与源端信息结合

LSTM 长短时记忆网络，通过门控机制 (gate network) 使循环神经网络能够对输入的信息进行选择，能够记住过去的信息，同时还能有挑选的遗忘模型认为不重要的信息，从而对长距离的信息依赖关系进行建模。受 LSTM 启发，将门控机制应用到我们的模型，来控制篇章信息和源端信息之间的选取比例。

对于位置 j 的当前输入源端隐层表示 $\mathbf{h}_j^x$，使用一个 gate network 来得到一个新的 $\mathbf{h}_j^x$，新的 $\mathbf{h}_j^x$ 同时包含当前隐层信息和相关联的篇章上文隐层信息，用 gate 权重来控制这两个信息所占比例：

$$\mathbf{h}_j^x = \lambda_j * \mathbf{h}_j^x + (1 - \lambda_j) * \mathbf{c}_j^d \tag{4.7}$$

gate network 的权重 λ_j 可以计算：

$$\lambda_j = \sigma(\mathbf{W}_i * \mathbf{h}_j^x + \mathbf{W}_s * \mathbf{c}_j^d)) \tag{4.8}$$

$\sigma(\cdot)$ 是一个 *sigmoid* 非线性函数，$\mathbf{W}_i$ 和 $\mathbf{W}_s$ 是模型参数。最终我们得到新的源端输入的隐层表示 $\mathbf{h}_j^x$。采用 gate network 是能够在篇章信息和当前句的信息中有权重的选取，选取权重可以由参数训练得到，这样既可以有效的利用到篇章信息也可以使得当前句子的翻译不会出现较大偏差。

4.5 实验

4.5.1 数据准备

在本章节，在中英翻译任务上对我们提出的模型进行验证，篇章机器翻译的训练语料是 TED 演讲数据集（TED Talks），语料主要来自于 International Workshop on Spoken Language Translation 国际口语机器翻译评测大赛 (IWSLT) 在 2014 年的评测活动公开的语料，TED 数据集中的数据主要是 TED 大会上的演讲稿，源自于由美国的非营利结构 TED Conference LLC. 组织。使用 TED dev2010 作为验证集来挑选模型，Ted tst2010，tst2011，tst2012，tst2013 作为测试集来评估模型最终的性能，四个测试集分别包含 1570，1245，1397，1261 句双语对。

4.5.2 训练细节

整个系统的基线模型是基于 Transformer 模型 (Vaswani 等, 2017b) 的，Transformer 模型在之前的章节有详细说明，这里采用的系统是开源工具 Fairseq-py (Edunov 等, 2017) 的实现。系统的基础参数完全依照 Vaswani 等 (2017b) 论文中提到的 Base 模型的参数来设置的。所有模型的训练都是在一个 8 个 NVIDIA TITAN Xp GPUs 卡的单服务器上训练的，batch size 设置为 4096 tokens，句长最长为 100。对于基线系统，训练 Step 达到 100k，每隔 1000 个 Step 保存一次模型参数，最后对最近的 5 个 Checkpoints 进行参数平均之后用测试集进行验证。

对于 TED 数据集，源端是基于 Char 的分词方法，目标端使用 16k 的 Byte Pair Encoding (BPE)(Sennrich 等, 2016) 操作数来进行数据集的字词分割。

对所有的系统采用同样的系统配置，词向量维度大小设置为 512，编码器和解码器的隐层单元的维度也被设置为 512，所有参数的初始化都使用范围是 $[-0.1, 0.1]$ 的均匀分布。优化算法采用 SGD(mini-batch stochastic gradient descent)，限制 batch 的大小为 4096 个字符。同时学习率的调整是通过 Adam 算法 (Kingma 和 Ba, 2015)($\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1e^{-6}$) 来优化的。Dropout rate 设置为 0.2，beam size 是 10。在解码过程中，设置 beam size 为 5，length penalty α=0.6 (Wu 等, 2016)。另外对所有系统使用 case-sensitive BLEU (Papineni 等, 2002) scores，最后的 BLEU 评价使用 *multi-bleu.pl*。

我们在第三章的 section **??**提到如何在干净的测试集中模拟同音异形字噪音和这两种结合的噪音，在本章的实验中采用同样构造对抗测试集 (含有噪音的测试集) 的方法，将替换概率 p 分别设置为 0.2，从而模拟真实的噪音环境。这么做的目的在于，当前没有公开的语音识别测试集，而干净的测试集是和语音识别

的识别结果有很大不同，因此在干净的测试集上添加噪音来验证我们模型的鲁棒性，这样得到了与干净测试集相对应的噪音测试集。与干净测试集的结果相对比，更全面的展现我们模型的性能。

4.5.3 对比系统

本章在以下系统做了实验，来对比我们系统的结果。

RNNSearch⋆ RNNSearch 系统的改进版，更多的细节可以通过这个链接了解[1].

Transformer Transformer 模型由谷歌提出的完全基于注意力机制的神经机器翻译系统，解码器编码器的序列建模都使用注意力函数代替循环神经网络。在翻译性能上超越 RNNSearch 模型。

Transformer+Doc 对 Transformer 模型在篇章信息上的改进，Transformer+Doc 模型是不含统一化语义空间的篇章翻译模型，在计算篇章信息的注意力向量时，Query 是汉字的隐层表示，其他模型的改进和我们的系统一样。

System	Clean Test Set				
	tst2010	**tst2011**	**tst2012**	**tst2013**	**Average**
RNNSearch	11.73	18.17	16.06	16.85	*15.70*
Transformer	13.86	20.08	17.73	18.38	*17.51*
Transformer+Doc	15.55	21.63	19.4	20.32	*19.225*
Our Model	**15.34**‡	**21.46**‡*	**19.22**‡*	**20.17**‡*	***19.05***

表 4.3 中英翻译任务在干净测试集上的实验结果

Table 4.3 Results of Chinese-English translation on clean test sets

4.5.4 实验结果

实验的结果在表 Table 4.3(“‡” 表示相比 RNNSearch⋆ 模型性能有明显提高，“⋆” 表示相比 Transformer 模型有明显提高。) 和表 Table 4.4(“‡” 表示相比 RNNSearch⋆ 模型性能有明显提高，“⋆” 表示相比 Transformer 模型有明显提高。) 进行了展示，其中表 Table 4.3是在干净的测试集上的结果，表 Table 4.4是我们构造的噪音测试集上的实验结果。

[1] https://github.com/nyu-dl/dl4mt-tutorial

System	Noise Test Set				
	tst2010	**tst2011**	**tst2012**	**tst2013**	**Average**
RNNSearch	10.13	16.22	14.14	15.05	*13.885*
Transformer	11.34	17.74	15.33	15.87	*15.07*
Transformer+Doc	13.78	18.43	17.13	18.14	*16.87*
Our Method	**14.07**‡	**20.18**‡*	**18.07**‡*	**18.99**‡*	***17.88***

表 4.4 中英翻译任务在噪音测试集上的实验结果

Table 4.4 Results of Chinese-English translation on noise test sets

在干净测试集上的结果，我们的系统与 Transformer 模型相比，在四个测试集上平均提高了 1.53 BLEU，这说明篇章信息的引入带来了性能的提升，篇章上下文能够对传统的翻译提供很强的辅助信息。另外，可以看到 Transformer+Doc 模型在干净的测试集上表现的相当优异，但是在噪音测试集上性能下降地非常明显，这说明当前先进的神经机器翻译系统在真实的噪音环境下的表现是不能让人满意的，这在之前的工作 Belinkov 和 Bisk (2018); Cheng 等 (2018) 也有被提到。而我们的模型在干净的测试集上同样表现优异，相比于 Transformer 并没有性能的下降，并且在噪音测试集上结果仅仅只比干净测试集稍差一点，比 Transformer 在噪音测试集上的结果要好非常多，结果要比 Transformer+Doc 模型要高 1.05 BLEU。这说明我们系统的鲁棒性更强，针对语音的识别错误表现出了很强的拟合能力。

System	p		
	0.1	0.2	0.3
Our Method	17.32	17.88	17.93

表 4.5 不同替换概率 p 的实验结果对比

Table 4.5 Results on different substitution probability p

另外，在表4.5可以看到篇章上文汉字替换拼音不同替换概率下的实验结果，在替换概率 $p = 0.2$ 和 $p = 0.3$ 要比 $p = 0.1$ 的结果更好，这说明不同程度的噪音采用对最终结果的影响较大。这个结论能够很好的指导我们在实际训练中如何训练一个更鲁棒的系统。

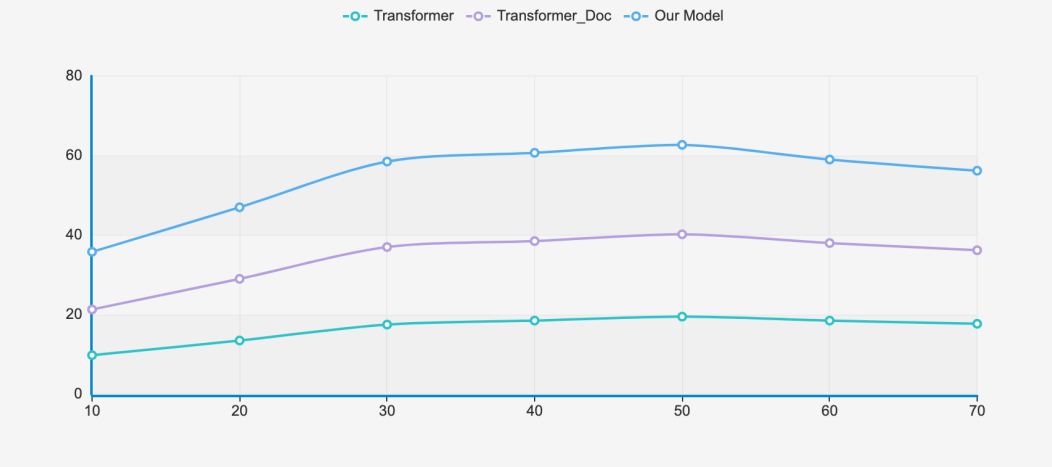

图 4.3 在不同源端句子长度的实验结果

Figure 4.3 Effect of source sentence lengths

4.5.5 句长测试

本章评估了我们模型在不同长度的源端句子下的表现，结果可以在图 4.3看到，可以很明显的看出无论是我们的模型还是基线系统，在句长 50 之前性能是一直提升的，但当句长超过 50 之后，翻译质量就有了下降，这个结论在之前的论文中也有提到 (Bahdanau 等, 2015)。

此外，句长曲线也表明长句子可以提供更多的上下文信息和更丰富的语音信息，这对噪声消歧是有帮助的。在每个长度间隔内，我们系统在所有噪音测试集上的性能都优于基线模型的。源句中噪音数量的增加并没有太大程度地降低我们模型的性能，说明了该方法对于噪音输入具有很强的鲁棒性。

4.6 本章小结

篇章翻译一般是用于解决一致性问题 (包括指代，时态等) 和歧义问题，从某种角度来看，噪音输入其实就是一种歧义问题，我们创新性的利用篇章信息来对噪音输入进行纠错来提高模型泛化能力，本章从两点探讨了噪音问题的解决，一个是如何从前几句的篇章句子中提取出需要的信息，二是如何利用提取的信息来对当前词进行改进和信息补充。实验结果证明了我们模型的有效性，模型方法能够很好地处理噪音输入，具有较强的鲁棒性。

第 5 章　总结与展望

5.1　总结

机器翻译是自然语言处理的上层任务，是自然语言处理技术的集成，从简单的分词，命名实体识别到复杂的句法词法分析和模型对齐等。而神经机器翻译在近两年来取得了突飞猛进的进步，各种模型不断被提出，性能效果不断被刷新。针对现有机器翻译模型的缺陷，提出针对性的改进，针对不同语言也有不同的特征来辅助翻译，比如汉语中的拼音这一重要特征被用来提高翻译鲁棒性，对机器翻译模型的改进具有非常重要的研究价值。

本文研究基于神经机器翻译系统进行改进，主要的工作内容如下：

1. 融合双端翻译历史信息的神经机器翻译

注意力机制 (Attention Mechanism) 极大地增强了传统的神经机器翻译的性能，注意力机制的引入使得模型可以在解码阶段有选择性地获得源语言句子的信息来生成目标词。然而，可以发现在传统的基于注意力的神经机器翻译中，目标端和源端的已翻译历史信息都没有被充分利用，这常常导致在计算注意力向量时出现对齐错误，尤其是在一些复杂的情况下，神经机器翻译是无法预测正确的译文。为了解决这一问题，本文提出了一种新的注意力机制 Bilingual History Involved Attention，融合双端翻译历史的注意力模型。该注意力机制维护两个向量来跟踪目标端已生成的历史信息和源端已被翻译的历史信息。本文提出的方法在汉英翻译任务，在 NIST 测试集上取得了 1.4 BLEU 值的性能提升，并且与先前的神经机器翻译结果相比显著提高了对齐质量。

2. 引入拼音信息的机器翻译对于语音识别输入的鲁棒性改进

在许多实际应用中，神经机器翻译系统必须处理来自自动语音识别系统 (ASR) 的输入，对于语音识别的结果可能包含一些噪音，尤其是在复杂的环境中产生的数据，经常会错误的生成一些词所对应的同音异形字和相近音异形字。这种情况往往会导致翻译性能急剧下降。在构建语音翻译系统时会产生两个很明显的问题，一是模型训练和系统测试时的不一致问题，二是由含有噪音的输入导致的翻译错误问题。本文创新性的提出了一种处理这两个问题的方法，以提高翻译时对语音识别错误输入的鲁棒性。首先对训练数据进行修改来模拟语音识别输出的错误类型，使训练和测试中的数据分布保持一致。其次，关注同音异形字和相近音异形字的自动语音识别错误，并利用它们的拼音信息帮助翻译模型

从含有噪音的错误输入中纠正过来。在两个汉英数据集上的实验表明，该方法对语音识别噪音输入更具鲁棒性，并能显著优于强基线系统。

3. 基于拼音信息的篇章级机器翻译对于噪音输入纠错的研究

篇章信息一般是用于解决机器翻译中存在的一致性问题(包括指代，时态等)和歧义问题。篇章翻译能够利用丰富的上下文信息来改进当前句子的翻译效果。对于语音识别场景下的机器翻译，语音识别的结果往往是含有噪音的，从某种角度来看，语音识别的噪音输入也是一种歧义问题。基于此，本文创新性的利用篇章信息来对噪音输入进行纠错来提高模型泛化能力，结合篇章翻译的做法，从篇章信息中提取出与当前噪音词相关的信息，然后利用提取的信息来对当前词进行改进和信息补充。针对篇章翻译的特点和真实环境下的噪音问题，本文提出了基于篇章信息来提高的神经机器翻译鲁棒性的方法，对整个系统的性能和鲁棒性都带来极大的提高。

5.2 展望

随着互联网的发展和国际间交流的增加，构建高质量的机器翻译系统是十分迫切的，机器翻译不但拥有着重要的研究价值还有着广阔的应用发展前景。当前的神经机器翻译在近几年虽然已经取得了令人欣喜的结果，但依然存在着许多问题，这些问题影响这机器翻译的实际应用和发展。目前主要的问题和未来的研究方向有：

1. 专有名词的翻译问题

错翻、漏翻和重复翻译是之前机器翻译中的普遍问题，我们的模型也探讨了这一问题并给出了出色的解决方案，但是对于人名、地名、组织机构名称，缩略语、俚语、成语等专业术语的翻译则受限于训练语料的局限和低词频带来的语义不明，导致机器翻译对于这些专有名词的翻译依然捉襟见肘。

2. 领域适应问题

机器翻译的高质量双语语料的获取成本是比较高，在实际应用中，往往无法获得大量的领域双语对齐语料。而利用传统的双语语料训练的系统在一些细分领域，比如政治，教育等领域的翻译效果会大幅降低。这给机器翻译的实际应用和商用带来了巨大的挑战。因此，研究如何借助大量的跨领域单语数据进行机器翻译也是一个重要的研究方向。

3. 机器翻译语义和语用的局限性

不同语言之间的句法差异是非常大的，像英汉语对，句法上的不同已经影响到语义了，机器翻译在翻译时是认为源端与目标端语义是对等的，而实际上语义不对等导致翻译对齐问题。双语句法差异导致的翻译译文逻辑混乱。对于多义词，机器翻译往往会选择高频词义，但词义则强烈依赖上下文语境，从而出现语义上的不对等，从而导致翻译错误问题。

参考文献

BAHDANAU D, CHO K, BENGIO Y. Neural machine translation by jointly learning to align and translate[C/OL]//3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings. 2015. http://arxiv.org/abs/1409.0473.

BAWDEN R, SENNRICH R, BIRCH A, HADDOW B. Evaluating discourse phenomena in neural machine translation[C/OL]//Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers). 2018: 1304-1313. https://aclanthology.info/papers/N18-1118/n18-1118.

BELINKOV Y, BISK Y. Synthetic and natural noise both break neural machine translation[C/OL]//6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings. 2018. https://openreview.net/forum?id=BJ8vJebC-.

BERARD A, BESACIER L, KOCABIYIKOGLU A C, PIETQUIN O. End-to-end automatic speech translation of audiobooks[C/OL]//2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada, April 15-20, 2018. 2018: 6224-6228. https://doi.org/10.1109/ICASSP.2018.8461690.

BROWN P F, COCKE J, PIETRA S A D, PIETRA V J D, JELINEK F, LAFFERTY J D, MERCER R L, ROOSSIN P S. A statistical approach to machine translation[J]. Computational linguistics, 1990, 16(2):79-85.

CHAN W, LANE I. On online attention-based speech recognition and joint mandarin character-pinyin training[C/OL]//Interspeech 2016, 17th Annual Conference of the International Speech Communication Association, San Francisco, CA, USA, September 8-12, 2016. 2016: 3404-3408. https://doi.org/10.21437/Interspeech.2016-334.

CHENG J, DONG L, LAPATA M. Long short-term memory-networks for machine reading[C/OL]//Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016. 2016: 551-561. http://aclweb.org/anthology/D/D16/D16-1053.pdf.

CHENG Y, TU Z, MENG F, ZHAI J, LIU Y. Towards robust neural machine translation[C/OL]//Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers. 2018: 1756-1766. https://aclanthology.info/papers/P18-1163/p18-1163.

CHIANG D. A hierarchical phrase-based model for statistical machine translation[C]//Proceedings

of the 43rd Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2005: 263-270.

CHO E, NIEHUES J, WAIBEL A. Nmt-based segmentation and punctuation insertion for real-time spoken language translation[J]. Proc. Interspeech, 2017:2645-2649.

CHO K, VAN MERRIENBOER B, GÜLÇEHRE Ç, BAHDANAU D, BOUGARES F, SCHWENK H, BENGIO Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation[C/OL]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL. 2014: 1724-1734. http://aclweb.org/anthology/D/D14/D14-1179.pdf.

DU J, WAY A. Pinyin as subword unit for chinese-sourced neural machine translation[C]//Irish Conference on Artificial Intelligence and Cognitive Science. 2017.

EDUNOV S, OTT M, GROSS S. https://github.com/pytorch/ fairseq[J]. 2017.

FENG Y, ZHANG S, ZHANG A, WANG D, ABEL A. Memory-augmented neural machine translation[C/OL]//Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017. 2017: 1390-1399. https://aclanthology.info/papers/D17-1146/d17-1146.

GARCIA E M, ESPANA-BONET C, VILLODRE L M. Document-level machine translation as a re-translation process[J]. Procesamiento del lenguaje natural, 2014, 53:103-110.

GONG Z, ZHANG M, ZHOU G. Cache-based document-level statistical machine translation[C]// Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2011: 909-919.

HARDMEIER C, FEDERICO M. Modelling pronominal anaphora in statistical machine translation [C]//IWSLT (International Workshop on Spoken Language Translation); Paris, France; December 2nd and 3rd, 2010. 2010: 283-289.

HOCHREITER S, SCHMIDHUBER J. Long short-term memory[J]. Neural computation, 1997, 9 (8):1735-1780.

JEAN S, LAULY S, FIRAT O, CHO K. Does neural machine translation benefit from larger context? [J/OL]. CoRR, 2017, abs/1704.05135. http://arxiv.org/abs/1704.05135.

KALCHBRENNER N, BLUNSOM P. Recurrent continuous translation models[C]//Proc. the 2013 Conference on Empirical Methods in Natural Language Processing. 2013: 1700-1709.

KINGMA D P, BA J. Adam: A method for stochastic optimization[C/OL]//3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings. 2015. http://arxiv.org/abs/1412.6980.

KITANO H. Speech-to-speech translation: a massively parallel memory-based approach: volume 260 [M]. Springer Science & Business Media, 2012.

KOCABIYIKOGLU A C, BESACIER L, KRAIF O. Augmenting librispeech with french translations:

A multimodal corpus for direct speech translation evaluation[C]//Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018. 2018.

KOEHN P, OCH F J, MARCU D. Statistical phrase-based translation[C]//Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1. Association for Computational Linguistics, 2003: 48-54.

KOEHN P, FEDERICO M, SHEN W, BERTOLDI N, BOJAR O, CALLISON-BURCH C, COWAN B, DYER C, HOANG H, ZENS R, et al. Open source toolkit for statistical machine translation: Factored translation models and confusion network decoding[C]//Final Report of the 2006 JHU Summer Workshop. 2006.

LIN Z, FENG M, DOS SANTOS C N, YU M, XIANG B, ZHOU B, BENGIO Y. A structured self-attentive sentence embedding[C/OL]//5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings. 2017. https://openreview.net/forum?id=BJC_jUqxe.

LIU Y, SUN M. Contrastive unsupervised word alignment with non-local features[C/OL]//AAAI'15: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence. AAAI Press, 2015: 2295-2301. http://dl.acm.org/citation.cfm?id=2886521.2886640.

LUONG T, PHAM H, MANNING C D. Effective approaches to attention-based neural machine translation[C/OL]//Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015. 2015: 1412-1421. http://aclweb.org/anthology/D/D15/D15-1166.pdf.

MATUSOV E, KANTHAK S, NEY H. Integrating speech recognition and machine translation: Where do we stand?[C]//Proc. ICASSP. 2006: 1217-1220.

MENG F, LU Z, LI H, LIU Q. Interactive attention for neural machine translation[C/OL]//COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan. 2016: 2174-2185. http://aclweb.org/anthology/C/C16/C16-1205.pdf.

MNIH V, HEESS N, GRAVES A, et al. Recurrent models of visual attention[C]//Advances in neural information processing systems. 2014: 2204-2212.

OCH F J. Minimum error rate training in statistical machine translation[C/OL]//Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics. Sapporo, Japan: Association for Computational Linguistics, 2003: 160-167. http://www.aclweb.org/anthology/P03-1021. DOI: 10.3115/1075096.1075117.

PAPINENI K, ROUKOS S, WARD T, ZHU W J. Bleu: a method for automatic evaluation of machine translation[C]//Proc. the 40th annual meeting on association for computational linguistics. 2002: 311-318.

PEITZ S, WIESLER S, NUSSBAUM-THOM M, NEY H. Spoken language translation using automatically transcribed text in training[C]//Proc. IWSLT. 2012.

POST M, KUMAR G, LOPEZ A, KARAKOS D, CALLISON-BURCH C, KHUDANPUR S. Improved speech-to-text translation with the fisher and callhome spanish–english speech translation corpus[C]//Proc. IWSLT. 2013.

RUIZ N, FEDERICO M. Assessing the impact of speech recognition errors on machine translation quality[J]. Association for Machine Translation in the Americas (AMTA), Vancouver, Canada, 2014:261-274.

RUIZ N, GAO Q, LEWIS W, FEDERICO M. Adapting machine translation models toward misrecognized speech with text-to-speech pronunciation rules and acoustic confusability[C]//Proc. Interspeech. 2015.

SCHUSTER M, PALIWAL K K. Bidirectional recurrent neural networks[J]. IEEE Transactions on Signal Processing, 1997, 45(11):2673-2681.

SENNRICH R, HADDOW B. Linguistic input features improve neural machine translation[C]//Proc. WMT. 2016.

SENNRICH R, HADDOW B, BIRCH A. Neural machine translation of rare words with subword units [C/OL]//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Berlin, Germany: Association for Computational Linguistics, 2016: 1715-1725. http://www.aclweb.org/anthology/P16-1162.

SERDYUK D, WANG Y, FUEGEN C, KUMAR A, LIU B, BENGIO Y. Towards end-to-end spoken language understanding[C/OL]//2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada, April 15-20, 2018. 2018: 5754-5758. https://doi.org/10.1109/ICASSP.2018.8461785.

SPERBER M, NEUBIG G, NIEHUES J, WAIBEL A. Neural lattice-to-sequence models for uncertain inputs[C/OL]//Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017. 2017: 1380-1389. https://aclanthology.info/papers/D17-1145/d17-1145.

SUTSKEVER I, VINYALS O, LE Q V. Sequence to sequence learning with neural networks[C]// Advances in neural information processing systems. 2014: 3104-3112.

TSVETKOV Y, METZE F, DYER C. Augmenting translation models with simulated acoustic confusions for improved spoken language translation[C]//Proc. ACL, 2014.

TU Z, LU Z, LIU Y, LIU X, LI H. Modeling coverage for neural machine translation[C/OL]// Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers. 2016. http://aclweb.org/anthology/P/P16/P16-1008.pdf.

VASWANI A, SHAZEER N, PARMAR N, USZKOREIT J, JONES L, GOMEZ A N, KAISER L U, POLOSUKHIN I. Attention is all you need[M/OL]//GUYON I, LUXBURG U V, BENGIO S,

WALLACH H, FERGUS R, VISHWANATHAN S, GARNETT R. Advances in Neural Information Processing Systems 30. Curran Associates, Inc., 2017a: 5998-6008. http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

VASWANI A, SHAZEER N, PARMAR N, USZKOREIT J, JONES L, GOMEZ A N, KAISER Ł, POLOSUKHIN I. Attention is all you need[C]//Advances in Neural Information Processing Systems. 2017b: 6000-6010.

VILAR D, XU J, LUIS FERNANDO D, NEY H. Error analysis of statistical machine translation output.[C]//Proc. LREC. 2006: 697-702.

VOITA E, SERDYUKOV P, SENNRICH R, TITOV I. Context-aware neural machine translation learns anaphora resolution[C/OL]//Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers. 2018: 1264-1274. https://aclanthology.info/papers/P18-1117/p18-1117.

WANG L, TU Z, WAY A, LIU Q. Exploiting cross-sentence context for neural machine translation[C/OL]//Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017. 2017: 2826-2831. https://aclanthology.info/papers/D17-1301/d17-1301.

WANG M, LU Z, LI H, LIU Q. Memory-enhanced decoder for neural machine translation[C/OL]//Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016. 2016: 278-286. http://aclweb.org/anthology/D/D16/D16-1027.pdf.

WANG M, XIE J, TAN Z, SU J, XIONG D, BIAN C. Neural machine translation with decoding history enhanced attention[C]//Proceedings of the 27th International Conference on Computational Linguistics. 2018: 1464-1473.

WEISS R J, CHOROWSKI J, JAITLY N, WU Y, CHEN Z. Sequence-to-sequence models can directly translate foreign speech[C/OL]//Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017. 2017: 2625-2629. http://www.isca-speech.org/archive/Interspeech_2017/abstracts/0503.html.

WENG R, HUANG S, ZHENG Z, DAI X, CHEN J. Neural machine translation with word predictions[C/OL]//Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017. 2017: 136-145. https://aclanthology.info/papers/D17-1013/d17-1013.

WU Y, SCHUSTER M, CHEN Z, LE Q V, NOROUZI M, MACHEREY W, KRIKUN M, CAO Y, GAO Q, MACHEREY K, et al. Google's neural machine translation system: Bridging the gap between human and machine translation[J]. arXiv preprint arXiv:1609.08144, 2016.

ZHAI F, ZHANG J, ZHOU Y, ZONG C. Tree-based translation without using parse trees[J]. Proceedings of COLING 2012, 2012:3037-3054.

ZHANG J, WANG M, LIU Q, ZHOU J. Incorporating word reordering knowledge into attention-

based neural machine translation[C]//Proc. the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): volume 1. 2017: 1524-1534.

ZHANG J, MATSUMOTO T. Improving character-level japanese-chinese neural machine translation with radicals as an additional input feature[C]//Asian Language Processing (IALP), 2017 International Conference on. IEEE, 2017: 172-175.

ZHOU L, ZHANG J, ZONG C. Look-ahead attention for generation in neural machine translation [C]//National CCF Conference on Natural Language Processing and Chinese Computing. Springer, 2017: 211-223.

致 谢

时间荏苒，三年研究生时光就要结束了。在即将毕业之际，我要向所有支持和帮助过我的老师和同学们表示由衷的感谢！对培养我的中国科学院计算技术研究所表示感谢！

首先我要衷心感谢我的导师冯洋副研究员，冯老师对待科研的态度，认真，细致透彻，敏锐的洞察力，对于问题思考的方式，都特别值得我学习。读研这几年来，冯老师无论是在学习、科研还是日常生活工作等许多方面都给了我非常大的帮助和指导，为我的研究学习工作的提供了很多帮助和建议，每次和冯老师的交谈都使我受益匪浅。在本篇论文的撰写过程中，对于文章的书写，章节的设置，内容的侧重点，论文的结构等许多方面都给予了非常中肯的修改建议，本篇文章的完成离不开冯老师的悉心指导。在整个研究生生涯，冯老师给予了我非常多的帮助，如何确定一个科研方向，怎么思考，如何去做，冯老师都很耐心细致的指导，使我进步很大。冯老师对于科研的严谨态度和在工作时表现出的敏锐洞察力和渊博的知识都影响我至今，我将以冯老师为榜样继续努力。

其次，我还要感谢刘群老师，姜文斌老师、赵秋野老师、赵红梅老师、刘琳老师。刘群老师为人谦和风趣，他开拓的研究思路和敏捷的思维都深深地影响着我。感谢姜文斌老师为我们营造的非常好的科研环境，他对科研的坚持和执着，对学生循循善诱，都深深的指引着我。感谢赵秋野老师，她严谨的科研态度和一丝不苟的学习态度都是我时时刻刻都铭记在心的。感谢赵红梅老师在工作上对我的巨大帮助，提供丰富实验的语料，还有对实验室项目的悉心整理和付出。感谢刘琳老师将我们的学习生活安排的井井有条，使我们在学习时没有后顾之忧。

感谢我在搜狗机器翻译组实习期间所有同事对我的帮助。十分感谢李响师兄在实习期间为我提供了良好的研究环境，在实习阶段对我悉心指导，在职业生涯方面给予我的帮助和建议。感谢在搜狗实习期间认识的各位同事。

十分感谢实验室的同学们，感谢张文、张金超、马青松、刘舒曼、张源，顾茂杰、刘毅，丁春发、胡稼伟等师兄师姐，感谢他们对我在学习时的疑惑细心答疑以及不求回报的帮助。感谢和我同时进入实验室学习的同窗好友李京谕同学，感谢王树根、谷舒豪、杨郑鑫、申磊、邵晨泽、单勇、李泽康等师弟师妹，实验室的发展离不开你们的努力，你们是实验室的未来！

感谢教育处卢文平、李琳、李丹、周世佳、冯刚和张平等几位老师在日常学

习生活和毕业的各个事项上给予我的帮助和关心。

特别感谢我的爸妈、亲人和其他陪伴我一路走来的朋友们，感谢你们在读研期间对我的支持。最后再次感谢所有支持帮助过我的人，谢谢。

作者简历

姓名：薛海洋　出生日期：1994.01.03　籍贯：山东省济宁市

教育情况

2016.9-2019.7　中国科学院计算技术研究所硕士研究生

2012.9-2016.7　山东大学本科生

攻读硕士期间参加的工程科研项目

2017.07-2018.01：苏州移动的翻译项目 (所级横向项目)

2017.10-2018.03：一路一带中阿双向翻译项目 (所级横向项目)

2018.03-2018.05：参加 CWMT 多语言翻译项目的评测，并获得第三名

联系方式

通讯地址：北京市海淀区科学院南路 6 号中国科学院计算技术研究所

邮编：100190

E-mail：xuehaiyang@ict.ac.cn